

# BAZE PODATAKA

## SQL SELECT (I dio)



Neđeljko Lekić

Irena Orović

[www.etf.ac.me](http://www.etf.ac.me)

[www.elektronika.t-com.me](http://www.elektronika.t-com.me)

# U OVOJ LEKCIJI

- SQL SELECT
  - WHERE klauzula
  - SELECT iz više tabela
  - Povezivanje tabela

# SQL SELECT: PREGLED

**SELECT**

**[DISTINCT | ALL] <column-list>**

**FROM <table-names>**

**[WHERE <condition>]**

**[ORDER BY <column-list>]**

**[GROUP BY <column-list>]**

**[HAVING <condition>]**

- (*[ ] - optional, | - or*)

# PRIMJERI TABELA

Student

ID	Ime	Prezime
103	Jovan	Simić
104	Marina	Jokić
105	Jana	Brkić
106	Marko	Jokić
107	Jovan	Brkić

Rezultati

ID	Kod	Bodovi
103	BPD	72
103	RHD	58
104	PR1	68
104	RHD	65
106	PR2	43
107	PR1	76
107	PR2	60
107	RHD	35

Predmeti

Kod	Naziv
BPD	Baze podataka
PR1	Programiranje 1
PR2	Programiranje 2
RHD	Računarski hardver

# DISTINCT i ALL

- Ponekad ne želite duplirane stavke

Upotrebom **DISTINCT**  
uklanjaju se duplikati.

```
SELECT DISTINCT  
Prezime  
FROM Student
```

Prezime
Simić
Brkić
Jokić

Upotrebom **ALL**  
ostaju – default.

```
SELECT ALL  
Prezime  
FROM Student
```

Prezime
Simić
Jokić
Brkić
Jokić
Brkić

# WHERE klauzula

- Obično se ne žele svi redovi u izvještaju
  - **WHERE** klauzula odabira redove (zapise) za izvještaj.
  - Ima oblik uslova – samo oni redovi koji zadovoljavaju uslov biti će odabrani.
- Primjeri uslova:
  - `Bodovi < 40`
  - `Ime = 'Jovan'`
  - `Ime <> 'Jovan'`
  - `Ime = Prezime`
  - `(Ime = 'Jovan') AND (Prezime = Simić')`
  - `(Bodovi < 40) OR (Bodovi > 70)`

# WHERE: PRIMJERI

```
SELECT * FROM Rezultati  
WHERE Bodovi >= 60
```

```
SELECT DISTINCT ID  
FROM Rezultati  
WHERE Bodovi >= 60
```

ID	Kod	Bodovi
103	BPD	72
104	PR1	68
104	RHD	65
107	PR1	76
107	PR2	60

ID
103
104
107

# WHERE: PRIMJER

- Data je tabela

Rezultati

ID	Kod	Bodovi
103	BPD	72
103	RHD	58
104	PR1	68
104	RHD	65
106	PR2	43
107	PR1	76
107	PR2	60
107	RHD	35

- Napisati SQL upit koji će naći listu ID brojeva i bodova za predmet sa kodom RHD, studenata koji su sakupili 40 ili više bodova.

ID	Bodovi
103	58
104	65



# JEDNO RJEŠENJE

Želimo samo ID i Bodove, Kod NE!

Jednostr. navodnici oko stringa

```
SELECT ID, Bodovi FROM Rezultati
```

```
WHERE (Kod = 'RHD') AND
```

```
(Bodovi >= 40)
```

Interesuju nas samo redovi sa kodom RHD

Interesuju nas samo redovi koji imaju više od 40 bodova.

# SELECT IZ VIŠE TABELA

- Često je potrebno kombinovati podatke iz dvije ili više tabele.
- Efekt proizvoda može se dobiti upotrebom:  
`SELECT * FROM Table1, Table2...`
- Ako tabele sadrže kolone istog imena - nedefinisan rezultat.
- Ovo se rješava označavanjem kolone imenom tabele:  
`TableName.Column`

# SELECT IZ VIŠE TABELA

**SELECT**

Ime, Prezime, Bodovi

**FROM** Student, Rezultati

**WHERE**

(Student.ID =  
Rezultati.ID)

**AND**

(Bodovi >= 40)

Student

ID	Ime	Prezime
103	Jovan	Simić
104	Marina	Jokić
105	Jana	Brkić

Rezultati

ID	Kod	Bodovi
103	BPD	72
103	RHD	58
104	PR1	68
104	RHD	65
106	PR2	43
107	PR1	76
107	PR2	60
107	RHD	35

# SELECT IZ VIŠE TABELA

```
SELECT * FROM Student, Rezultati
```

ID	Ime	Prezime	ID	Kod	Bodovi
103	Jovan	Simić	103	BPD	72
103	Jovan	Simić	103	RHD	58
103	Jovan	Simić	104	PR1	68
103	Jovan	Simić	104	RHD	65
103	Jovan	Simić	106	PR2	43
103	Jovan	Simić	107	PR1	76
103	Jovan	Simić	107	PR2	60
103	Jovan	Simić	107	RHD	35
104	Marina	Jokić	103	BPD	72
104	Marina	Jokić	103	RHD	58
104	Marina	Jokić	104	PR1	68
104	Marina	Jokić	104	RHD	65
104	Marina	Jokić	104	PR2	43

su upareni  
sa prvim  
redom  
iz tebele  
Student ...

I zatim sa  
drugim...

I tako  
dalje

Svi redovi iz  
tebele Rezultati

# SELECT IZ VIŠE TABELA

```
SELECT * FROM Student, Rezultati  
WHERE (Student.ID = Rezultati.ID)
```

ID	Ime	Prezime	ID	Kod	Bodovi
103	Jovan	Simić	103	BPD	72
103	Jovan	Simić	103	RHD	58
103	Jovan	Simić	104	PR1	68
103	Jovan	Simić	104	RHD	65
103	Jovan	Simić	106	PR2	43
103	Jovan	Simić	107	PR1	76
103	Jovan	Simić	107	PR2	60
103	Jovan	Simić	107	RHD	35
104	Marina	Jokić	103	BPD	72
104	Marina	Jokić	103	RHD	58
104	Marina	Jokić	104	PR1	68
104	Marina	Jokić	104	RHD	65
104	Marina	Jokić	104	RHD	65

# SELECT IZ VIŠE TABELA

```
SELECT * FROM Student, Rezultati  
WHERE (Student.ID = Rezultati.ID)
```

ID	Ime	Prezime	ID	Kod	Bodovi
103	Jovan	Simić	103	BPD	72
103	Jovan	Simić	103	RHD	58
104	Marina	Jokić	104	PR1	68
104	Marina	Jokić	104	RHD	65
106	Marko	Jokić	106	PR2	43
107	Jovan	Brkić	107	PR1	76
107	Jovan	Brkić	107	PR2	60
107	Jovan	Brkić	107	RHD	35

Student.ID

Rezultati.ID

# SELECT IZ VIŠE TABELA

```
SELECT * FROM Student, Rezultati  
WHERE (Student.ID = Rezultati.ID) AND (Bodovi >= 40)
```

ID	Ime	Prezime	ID	Kod	Bodovi
103	Jovan	Simić	103	BPD	72
103	Jovan	Simić	103	RHD	58
104	Marina	Jokić	104	PR1	68
104	Marina	Jokić	104	RHD	65
106	Marko	Jokić	106	PR2	43
107	Jovan	Brkić	107	PR1	76
107	Jovan	Brkić	107	PR2	60
107	Jovan	Brkić	107	RHD	35

# SELECT IZ VIŠE TABELA

```
SELECT * FROM Student, Rezultati  
WHERE (Student.ID = Rezultati.ID) AND (Bodovi >= 40)
```

ID	Ime	Prezime	ID	Kod	Bodovi
103	Jovan	Simić	103	BPD	72
103	Jovan	Simić	103	RHD	58
104	Marija	Jokić	104	PR1	68
104	Marija	Jokić	104	RHD	65
106	Marko	Jokić	106	PR2	43
107	Jovan	Brkić	107	PR1	76
107	Jovan	Brkić	107	PR2	60



# SELECT IZ VIŠE TABELA

```
SELECT Ime, Prezime, Bodovi FROM Student, Rezultati  
WHERE (Student.ID = Rezultati.ID) AND (Bodovi >= 40)
```

Ime	Prezime	Bodovi
Jovan	Simić	72
Jovan	Simić	58
Marija	Jokić	68
Marija	Jokić	65
Marko	Jokić	43
Jovan	Brkić	76
Jovan	Brkić	60

# SELECT IZ VIŠE TABELA

```
SELECT Ime, Prezime, Bodovi FROM Student, Rezultati  
WHERE (Student.ID = Rezultati.ID) AND (Bodovi >= 40)
```

Ime	Prezime	Bodovi
Jovan	Simić	72
Jovan	Simić	58
Marija	Jokić	68
Marija	Jokić	65
Marko	Jokić	43
Jovan	Brkić	76
Jovan	Brkić	60

# SELECT IZ VIŠE TABELA

- Prilikom selekcije iz više tabela gotovo uvijek se koristiti **WHERE** klauzula za nalaženje zapisa sa zajedničkom vrijednošću.

```
SELECT * FROM
    Student,
    Rezultati,
    Predmeti
WHERE
    Student.ID =
    Rezultati.ID
AND
    Predmeti.Kod =
    Rezultati.Kod
```

# SELECT IZ VIŠE TABELA

Student			Rezultati			Predmet	
ID	Ime	Prezime	ID	Kod	Bodovi	Kod	Naziv
103	Jovan	Simić	103	BPD	72	BPD	Baze podataka
103	John	Simić	103	RHD	58	RHD	Računarski hardver
104	Marija	Jokić	104	PR1	68	PR1	Programiranje 1
104	Marija	Jokić	104	RHD	65	RHD	Računarski hardver
106	Marko	Jokić	106	PR2	43	PR2	Programiranje 2
107	Jovan	Brkić	107	PR1	76	PR1	Programiranje 1
107	Jovan	Brkić	107	PR2	60	PR2	Programiranje 2
107	Jovan	Brkić	107	RHD	35	RHD	Računarski hardver

Student.ID = Rezultati.ID

Rezultati.Kod = Predmet.Kod

# SPAJANJE TABELA (JOIN)

■ JOIN – može se koristiti za kombinovanje tabela.

- Postoji više načina spajanja
  - CROSS JOIN
  - INNER JOIN
  - NATURAL JOIN
  - OUTER JOIN
- OUTER JOIN je povezivanje sa NULL-ama – više o tome kasnije

## A CROSS JOIN B

- vraća uparene, svaki sa svakim, redove iz A i B.

## A NATURAL JOIN B

- vraća parove redova sa istom vrijednosti u koloni istog imena, i bez dupliranja kolona.

## A INNER JOIN B

- Vraća parove redova koji zadovoljavaju uslov.

# CROSS JOIN

```
SELECT * FROM  
Student CROSS JOIN  
Upis
```

Student

ID	Ime
123	Jovan
124	Marija
125	Marko
126	Jana

Upis

ID	Kod
123	BPD
124	PRG
124	BPD
126	PRG

ID	Ime	ID	Kod
123	Jovan	123	BPD
124	Marija	123	BPD
125	Marko	123	BPD
126	Jana	123	BPD
123	Jovan	124	PRG
124	Marija	124	PRG
125	Marko	124	PRG
126	Jana	124	PRG
123	Jovan	124	BPD
124	Marija	124	BPD

# NATURAL JOIN

Student

ID	Ime
123	Jovan
124	Marija
125	Marko
126	Jana

```
SELECT * FROM  
Student NATURAL  
JOIN Upis
```

Upis

ID	Kod
123	BPD
124	PRG
124	BPD
126	PRG

ID	Ime	Kod
123	Jovan	BPD
124	Marija	PRG
124	Marija	BPD
126	Jana	PRG

# CROSS i NATURAL JOIN

```
SELECT * FROM  
  A CROSS JOIN B
```

- je isto što i

```
SELECT * FROM A, B
```

```
SELECT * FROM  
  A NATURAL JOIN B
```

- je isto što i

```
SELECT A.col1,... A.coln,  
[i sve druge kolone  
osim B.col1,... B.coln]
```

```
FROM A, B
```

```
WHERE A.col1 = B.col1
```

```
AND A.col2 = B.col2 ...
```

```
AND A.coln = B.coln
```

(podrazumijeva se da  
col1... coln u A i B  
imaju isto ime)



# INNER JOIN

- **INNER JOINS**

specificira uslov koji parovi redova zadovoljavaju

```
SELECT * FROM  
A INNER JOIN B  
ON <condition>
```

- Može se koristiti i

```
SELECT * FROM  
A INNER JOIN B  
USING  
    (col1, col2,...)
```

- Povezuje redove kod kojih su vrijednosti datih kolona jednake

# INNER JOIN

Student

ID	Ime
123	Jovan
124	Marija
125	Marko
126	Jana

```
SELECT * FROM  
Student INNER JOIN  
Upis USING (ID)
```

Upis

ID	Kod
123	BPD
124	PRG
124	BPD
126	PRG

ID	Ime	ID	Kod
123	Jovan	123	BPD
124	Marija	124	PRG
124	Marija	124	BPD
126	Jana	126	PRG

# INNER JOIN

Kupci

Ime	Budzet
Simić	100,000
Jović	150,000
Grgić	80,000

```
SELECT * FROM  
Kupci INNER JOIN  
Imovine ON  
Cijena <= Budzet
```

Imovine

Adresa	Cijena
15 Slobode	85,000
12 Njegoševa	125,000
87 Čopićeva	175,000

Ime	Budzet	Adresa	Cijena
Simić	100,000	15 Slobode	85,000
Jović	150,000	15 Slobode	85,000
Jović	150,000	12 Njegoševa	125,000

# INNER JOIN

```
SELECT * FROM  
A INNER JOIN B  
ON <condition>
```

■ je isto kao i

```
SELECT * FROM A, B  
WHERE <condition>
```

```
SELECT * FROM  
A INNER JOIN B  
USING (col1, col2, ...)
```

■ je isto kao i

```
SELECT * FROM A, B  
WHERE A.col1 = B.col1  
AND A.col2 = B.col2  
AND ...
```

# JOIN i WHERE klauzule

- JOIN (izgleda) nije neophodno
  - Može se dobiti isti efekat selektovanjem iz više tabela sa odgovarajućom WHERE klauzulom
  - Pa, bismo li koristili JOIN ili ne?
- Da, jer
  - JOIN ponekad vodi konciznijim upitima
  - NATURAL JOIN je prilično čest slučaj spajanja tabela
- Ne, jer
  - Podrška za JOIN varira između SQL-a različitih DBMS-ova.

# PISANJE UPITA

- Kod pisanja upita
  - Često postoje mnogi načini da se napiše upit.
  - Za sada treba voditi računa da upit bude tačan, jasan, i koncizan, tim redom.
  - Ne treba previše brinuti o domišljatosti i efikasnosti.
- Mnogi DBMS-ovi imaju optimizator upita.
  - Oni analiziraju korisnikov upit i procjenjuju koliko se efikasno izvršava.
  - Jednostavni upit je lakše optimizovati.
  - Kanije ćemo se upoznati sa nekim načinima povećanja efikasnosti upita.

# ZA VJEŽBU

Koristeći tabele sa vježbi iz predhodne lekcije, sastaviti sljedeće upite:

Numere

cID	Num	Naziv	Traj	uID
1	1	Violent	239	1
1	2	Every Girl	410	1
1	3	Breather	217	1
1	4	Part of Me	279	1
2	1	Star	362	1
2	2	Teaboy	417	2

CD

cID	Naslov	Cijena
1	Mix	9.99
2	Kompilacija	12.99

Umjetnici

uID	Ime
1	Stellar
2	Cloudboy

- 1.** Napisati upit koji vraća listu svih CD naslova.
- 2.** Napisati upit koji vraća listu naziva numera koje traju duže od 300 sekundi.
- 3.** Napisati upit koji vraća listu imena umjetnika koji imaju numeru čije je trajanje duže od 400 sekundi.

## ZA VJEŽBU

4. Napisati upit koji vraća listu imena umjetnika koji imaju numeru na "Kompilacija" CD-u.
5. Napraviti upit pretraživanja nad relacijama Umjetnici, Numere i CD koji će kao rezultat vratiti ime umjetnika, ime Numere koju umjetnik izvodi i naslov CD-a kojemu numera pripada.
6. Napraviti upit pretraživanja nad relacijama Umjetnici, Numere i CD koji će kao rezultat vratiti naziv Numere, njeno trajanje i naslov CD-a kojemu numera pripada, ukoliko je njihov autor umjetnik sa imenom "Steller".



# SLJEDEĆA LEKCIJA

- Još o SQL SELECT
  - Pseudonimi (lažna imena)
  - 'Spajanje sa sobom'
  - Podupiti
  - IN, EXISTS, ANY, ALL