

Fakultet tehničkih nauka, Novi Sad

Predmet:

## **Baze podataka**

dr Slavica Kordić

dr Vladimir Ivančević

Marija Kukić

Marko Vještica

Nikola Todorović

1

## **Rad u učionici (1/2)**

- Baze podataka (db2016):
  - Studentska korisnička šema (user schema)
  - pod nazivom **swXY**
    - **Username:** swXY
    - **Password:** ftn
  - gde je X broj indeksa, a Y godina upisa

2

## Rad u učionici (2/2)

- Podaci potrebni za konektovanje na bazu podataka (db2016):
  - Host Name:
    - 192.168.0.102 (za laboratoriju)
    - localhost
  - Port Number:
    - 1522 (za laboratoriju)
    - 1521
  - Oracle SID, database name:
    - db2016 (za laboratoriju)
    - xe

3

## SQL (Structured Query Language)

- **SQL**
  - standardni jezik relacionih sistema za upravljanje bazama podataka
  - jezik visokog nivoa deklarativnosti
  - objedinjuje funkcije jezika za definiciju podataka, jezik za manipulaciju podacima i upitni jezik

4

4

## **SQL (Structured Query Language)**

- **Namena i zadaci SQL-a u okviru sistema za upravljanje bazama podataka**
  - administratorima baze podataka za obavljanje poslova administracije
  - programerima za izradu aplikacija nad bazom podataka
  - krajnjim korisnicima, za postavljanje upita nad bazom podataka

5

5

## **SQL (Structured Query Language)**

- **SQL se javlja u formama:**
  - interaktivnog jezika sistema za upravljanje bazama podataka
  - ugrađenog jezika u jezik III generacije
  - sastavnog dela jezika IV generacije

6

6

## SQL (Structured Query Language)

- **Saglasno nameni i vrstama korisnika koji ga upotrebljavaju, SQL obezbeđuje realizaciju sledećih zadataka:**
  - izražavanje upita putem upitnog jezika (naredba SELECT)
  - ažuriranje baze podataka putem jezika za manipulaciju podacima (naredbe INSERT, DELETE i UPDATE)
  - realizacija implementacione šeme baze podataka i definisanje fizičke organizacije baze podataka (naredbe CREATE, DROP i ALTER)
  - automatsko održavanje rečnika podataka

7

7

## SQL (Structured Query Language)

- **Saglasno nameni i vrstama korisnika koji ga upotrebljavaju, SQL obezbeđuje realizaciju sledećih zadataka:**
  - transakcijska obrada podataka ( naredbe COMMIT, ROLLBACK, SAVEPOINT)
  - zaključavanje resursa (naredba LOCK TABLE)
  - zaštita podataka od neovlašćenog pristupa (naredbe GRANT, REVOKE)
  - praćenje zauzeća resursa i performansi rada sistema za upravljanje bazama podataka (naredbe AUDIT, EXPLAIN PLAN)
  - obezbeđenje proceduralnog načina obrade podataka "slog po slog" (naredbe za rad sa kursorom: OPEN, FETCH, CLOSE)

8

8

## SQL (Structured Query Language)

- Sintaksa SQL-a zavisi od proizvođača sistema za upravljanje bazama podataka.

9

9

## Primer

radnik({Mbr, Ime, Prz, Sef, Plt, God,Pre}, {Mbr}),  
 projekat({Spr, Ruk, Nap, Nar}, {Spr}),  
 radproj({Spr, Mbr, Brc}, {Spr + Mbr}),

radnik[Sef]  $\subseteq$  radnik[Mbr],  
 projekat[Ruk]  $\subseteq$  radnik[Mbr],  
 radproj[Mbr]  $\subseteq$  radnik[Mbr],  
 radproj[Spr]  $\subseteq$  projekat[Spr].

10

10

## Tabela radnik

- **Mbr** - maticni broj radnika
- **Ime** - ime radnika
- **Prz** - prezime radnika
- **Sef** - maticni broj direktno nadređenog rukovodioca - radnika
- **Plt** - mesečni iznos plate radnika
- **God** - Datum rođenja radnika
- **Pre** – godišnja premija na platu radnika

Obeležja Mbr, Ime, Prz ne smeju imati null vrednost.  
Plata ne sme biti manja od 500

11

11

## Tabela radnik

```
CREATE TABLE radnik(
    Mbr integer NOT NULL,
    Ime varchar(20) NOT NULL,
    Prz varchar(25) NOT NULL,
    Sef integer,
    Plt decimal(10, 2),
    Pre decimal(6, 2),
    God date NOT NULL,
    CONSTRAINT radnik_PK PRIMARY KEY (Mbr),
    CONSTRAINT radnik_FK FOREIGN KEY (Sef)
        REFERENCES Radnik (Mbr),
    CONSTRAINT radnik_CH CHECK (Plt >= 500)
);
```

12

12

## Tabela projekat

- **Spr** - sifra projekta
- **Ruk** - rukovodilac projekta
- **Nap** - naziv projekta
- **Nar** - narucilac projekta

Obeležja Spr i Ruk ne smeju imati null vrednost, dok obeležje Nap mora imati jedinstvenu vrednost

13

13

## Tabela projekat

```
CREATE TABLE projekat
(
    Spr integer not null,
    Ruk integer not null,
    Nap varchar(30),
    Nar varchar(30),
    CONSTRAINT projekat_PK PRIMARY KEY (Spr),
    CONSTRAINT projekat_FK FOREIGN KEY (Ruk)
        REFERENCES Radnik (Mbr),
    CONSTRAINT projekat_UK UNIQUE (Nap)
);
```

14

14

## Tabela radproj

- **Spr** - sifra projekta
- **Mbr** - maticni broj radnika
- **Brc** - broj casova nedeljnog angazovanja na projektu

Sva tri obeležja ne smeju da imaju null vrednost

15

15

## Tabela radproj

```
CREATE TABLE radproj
(
    Spr integer NOT NULL,
    Mbr integer NOT NULL,
    Brc integer NOT NULL,
    CONSTRAINT radproj_PK PRIMARY KEY (Spr, Mbr),
    CONSTRAINT radproj_rad_FK FOREIGN KEY (Mbr)
        REFERENCES radnik(Mbr),
    CONSTRAINT radproj_prj_FK FOREIGN KEY (Spr)
        REFERENCES projekat(Spr)
);
```

16

16



## Kreiranje tabele

```
CREATE TABLE [šema.]<naziv_tabele>
(<naziv_kolone> <tip_podatka> [DEFAULT
izraz] [, ...]
CONSTRAINT <naziv_ogranicenja>
<definicija_ogranicenja> [, ...]);
```

- šema – poklapa se sa nazivom korisnika
- **DEFAULT opcija:**
  - Specificira se predefinisana vrednost za kolonu, koja se koristi ukoliko se prilikom ubacivanja podataka izostavi vrednost za tu kolonu

17

17

## Naziv tabele i kolone

- mora početi slovom,
- mora biti između 1 i 30 znakova dužine,
- mora sadržati samo velika i mala slova, cifre, \_, \$ i #,
- ne sme se poklapati sa nazivom nekog drugog objekta koji je kreirao isti korisnik,
- ne sme biti rezervisana reč Oracle servera.
- Nazivi nisu case sensitive.

18

18

## SQL tipovi podataka

Tip podatka	Opis
– <b>VARCHAR2(size)</b>	niz karaktera promenljive dužine, maksimalne dužine <i>size</i> ; minimalna dužina je 1, maksimalna je 4000
– <b>CHAR(size)</b>	Niz karaktera fiksne dužine od <i>size</i> bajtova; default i minimalna dužina je 1, maksimalna dužina je 2000
– <b>NUMBER(p,s)</b>	broj ukupnog broja cifara p, od čega je s cifara iza decimalnog zareza; p može imati vrednosti od 1 do 38
– <b>DATE</b>	vrednosti za vreme i datum
– <b>LONG</b>	niz karaktera promenljive dužine do 2 GB – za kompatibilnost sa starijim verzijama Oracle-a
– <b>CLOB</b>	niz karaktera promenljive dužine do 4 GB
– <b>BLOB</b>	binarni podaci do 4 GB
– <b>BFILE</b>	binarni podaci smešteni u eksternom fajlu do 4 GB
– <b>ROWID</b>	jedinstvena adresa vrste u tabeli

19

19

## Izražavanje upita i osnovna struktura naredbe SELECT

- Sve vrste upita se u SQL-u izražavaju putem naredbe SELECT. Osnovna struktura SELECT naredbe je:

```
SELECT *| <lista_obeležja>  
FROM <lista_tabela>  
[WHERE <uslov_selekcije>]
```

<lista\_obeležja> sadrži obeležja nad kojima se formira rezultat upita, <lista\_tabela> sadrži nazive tabela potrebne za realizaciju upita, <uslov\_selekcije> izražava uslov selekcije podataka iz tabela koje su navedene iza službene reči FROM

20

20

## Upiti nad jednom tabelom

- Izlistati sadržaj svih tabela.

```
select * from radnik;  
select * from projekat;  
select * from radproj;
```

21

21

## Upiti nad jednom tabelom

- Prikazati imena i prezimena svih radnika.

```
select ime, prz  
from radnik;
```

22

22

## DISTINCT

```
SELECT [DISTINCT] <lista_obeležja>  
FROM <lista_tabela>  
WHERE <uslov_selekcije>
```

- Izlistati različita imena radnika.

```
select distinct ime from radnik;
```

23

23

## WHERE <uslov\_selekcije>

- Izlistati mbr, ime i prezime radnika koji imaju platu veću od 25000.

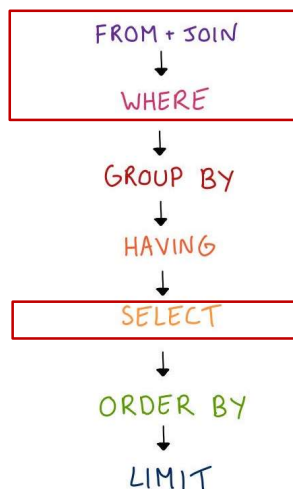
```
select mbr, ime, prz  
from radnik  
where plt>25000;
```

24

24

## Redosled izvršavanja klauzula

Redosled izvršavanja SQL klauzula



25

25

## Aritmetički izrazi

```

SELECT *|[DISTINCT] <lista_izraza>
FROM <lista_tabela>
WHERE <uslov_selekcije>
  
```

- Izlistati godišnju platu svakog radnika.

```

select mbr, ime, prz, plt*12
from radnik;
  
```

26

26

## **Null vrednost**

**x IS NULL – x je nula vrednost**

**x IS NOT NULL – x nije nula vrednost**

- Izlistati mbr, ime, prz radnika koji nemaju šefa.

**SELECT mbr, ime, prz from radnik  
where sef is null;**

27

27

## **BETWEEN**

- Izlistati mbr, ime, prz radnika čija je plata između 20000 i 24000 dinara.

**select mbr, ime, prz  
from radnik  
where plt between 20000 and 24000;**

28

28

## BETWEEN

- Izlistati ime, prz, god radnika rođenih između 1953 i 1975.

```
select ime, prz, god
from radnik
where god between '01-jan-1953' and '31-dec-1975';
```

- Razlika između between i < >?

29

29

## NOT BETWEEN

- Izlistati ime, prz, god radnika koji nisu rođeni između 1953 i 1975.

```
select ime, prz, god
from radnik
where god not between '01-jan-1953' and '31-dec-1975';
```

30

30

## LIKE

**<obeležje> LIKE <uzorak>**

- Izlistati mbr, ime, prz radnika čije prezime počinje na slovo M.

```
select mbr, ime, prz  
from radnik  
where prz like 'M%';
```

31

31

## NOT LIKE

- Izlistati mbr, ime, prz radnika čije ime ne počinje slovom A.

```
select mbr, ime, prz  
from radnik  
where ime not like 'A%';
```

32

32



## LIKE

- Izlistati mbr, ime, prz radnika čije ime sadrži slovo **a** na drugoj poziciji.

```
select mbr, ime, prz  
from radnik  
where ime like '_a%';
```

33

33

## LIKE

- Izlistati imena radnika koja počinju na slovo E. Imena ne bi trebalo da se ponavljaju.

```
select distinct ime  
from radnik  
where ime like 'E%';
```

34

34

## LIKE

- Izlistati radnike koji u svom imenu imaju slovo E (e).

```
select mbr, ime, prz  
from radnik  
where ime like '%e%' or ime like  
'%E%';
```

35

35

## IN

- Izlistati matične brojeve radnika koji rade na projektima sa šifrom 10, 20 ili 30.

```
select distinct mbr  
from radproj  
where spr in (10, 20, 30);
```

36

36

## IN

- Izlistati matične brojeve radnika koji rade na projektu sa šifrom 10 ili rade 2, 4, ili 6 sati.

```
select distinct mbr  
from radproj  
where brc in (2, 4, 6) or spr='10';
```

37

37

## NOT IN

- Izlistati matične brojeve radnika koji se ne zovu Ana ili Sanja.

```
select mbr, ime, prz  
from radnik  
where ime not in ('Ana', 'Sanja');
```

38

38

## Uređivanje izlaznih rezultata

```
SELECT *| <lista_izraza>  
FROM <lista_tabela>  
WHERE <uslov_selekcije>  
ORDER BY <podlista_obeležja>
```

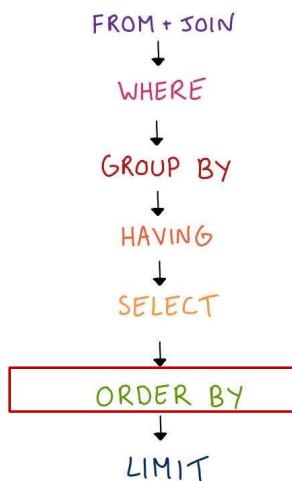
**ORDER BY** je uvek poslednja klauzula naredbe **SELECT**

39

39

## Redosled izvršavanja klauzula

Redosled izvršavanja SQL klauzula



40

40

## ORDER BY

- Prikazati radnike koji imaju šefa sortirano po prezimenu.

```
select mbr, ime, prz, plt  
from radnik  
where sef is not null  
order by prz asc;
```

41

41

## ORDER BY

- Neki primeri upotrebe klauzule ORDER BY.

```
SELECT Mbr, Ime, Prz, Plt  
FROM Radnik ORDER BY Prz, Ime;
```

```
SELECT Mbr, Prz, Ime, Plt  
FROM Radnik ORDER BY Prz ASC, Ime ASC;
```

```
SELECT Mbr, Prz, Ime, Plt  
FROM Radnik ORDER BY Prz ASC, Ime DESC;
```

42

42

## ORDER BY

- Neki primeri upotrebe klauzule ORDER BY.

```
SELECT Mbr, Prz, Ime  
FROM Radnik  
ORDER BY 2, 3, Plt;
```

```
SELECT Mbr, Prz, Ime  
FROM Radnik  
ORDER BY 2, 3, Plt * 1.17;
```

43

43

## ORDER BY

- Prikazati matične brojeve, imena, prezimena i plate radnika, po opadajućem redosledu iznosa plate.

```
SELECT Mbr, Ime, Prz, Plt Plata  
FROM Radnik  
ORDER BY Plata DESC;
```

44

44

## Uređivanje izlaznih rezultata

- Prikazati matične brojeve, **spojena** (konkatenirana) imena i prezimena radnika, kao i plate, uvećane za 17%.

```
SELECT Mbr,  
       Ime || ' ' || Prz "Ime i prezime",  
       Plt * 1.17 Plata  
FROM Radnik;
```

- **Concat funkcija**

45

45

## Zadatak

- Prikazati radnike čije prezime sadrži ime.  
Na primer **Marko Marković**, ili **Djordje Karadjordjevic**  
– Funkcije UPPER, LOWER

```
SELECT * from radnik where LOWER(prz)  
LIKE '%' || LOWER(ime) || '%'
```

46

46

## ANY

**$x \Theta \text{ ANY } (<\text{lista\_vrednosti}>)$**

**$\Theta \in \{<, >, <=, >=, !=, =\}$**

Primer:

**$x = \text{ANY } (<\text{lista\_vrednosti}>)$**

**$x$  je jednako makar jednoj vrednosti u  
 $<\text{listi\_vrednosti}>$**

47

47

## ANY

- Prikazati matične brojeve radnika, imena i prezimena i platu radnika koji se zovu Pera ili Moma.

```
SELECT Mbr, Ime, Prz, Plt
FROM Radnik
WHERE Ime = ANY ('Pera', 'Moma');
```

48

48



## ALL

**$x \Theta \text{ ALL } (<\text{lista\_vrednosti}>)$**

**$\Theta \in \{<, >, <=, >=, !=, =\}$**

Primer:

**$x != \text{ALL } (<\text{lista\_vrednosti}>)$**

**$x$  je različito od svake vrednosti u  
 $<\text{listi\_vrednosti}>$**

49

49

## ALL

- Prikazati matične brojeve radnika, imena i prezimena i platu radnika koji se ne zovu Pera ili Moma.

```
SELECT Mbr, Ime, Prz, Plt
FROM Radnik
WHERE Ime !=ALL ('Pera', 'Moma');
```

50

50

## Upotreba skupovnih funkcija

- Prikazati matične brojeve radnika, kao i plate, uvećane za NULL vrednost.

**SELECT Mbr, Plt + NULL FROM Radnik;**

- Prikazati matične brojeve radnika, kao i plate, uvećane za godišnju premiju.

**SELECT Mbr, Plt + Pre FROM Radnik;**

51

51

## Funkcija NVL(*izraz, konstanta*)

- Prikazati matične brojeve radnika, kao i plate, uvećane za godišnju premiju. Ukoliko za nekog radnika vrednost premije ne postoji, smatrati da ona iznosi 0.

**SELECT Mbr, Plt + NVL(Pre, 0)  
FROM Radnik;**

52

52

## Funkcija count

- COUNT(\*) – vraća ukupan broj selektovanih torki
- COUNT(<obeležje>) – vraća ukupan broj selektovanih torki, za koje vrednost <obeležja> nije nula vrednost
- COUNT(DISTINCT <obeležje>) – vraća ukupan broj različitih torki, za koje vrednost <obeležja> nije nula vrednost

53

53

## Funkcija count

- Koliko ima radnika?

```
select count(*) from radnik;
```

- Koliko ima šefova?

```
select count(distinct sef) broj_sefova  
from radnik;
```

54

54

## Funkcije max i min

- MAX(<obeležje>) – vraća maksimalnu vrednost za <obeležje>, uzimajući u obzir sve selektovane torke
- MIN(<obeležje>) – vraća minimalnu vrednost za <obeležje>, uzimajući u obzir sve selektovane torke

55

55

## Funkcije max i min

- Prikazati minimalnu i maksimalnu platu radnika.

```
select min(plt) minimalna, max(plt) maksimalna  
from radnik;
```

56

56

## Funkcija sum

- SUM(<obeležje>) – vraća zbir vrednosti datog <obeležja>, za sve selektovane torke, uključujući višestruko ponavljanje istih torki
- SUM(DISTINCT <obeležje>) – vraća zbir vrednosti datog <obeležja>, za sve različite selektovane torke

57

57

## Funkcija sum

- Prikazati broj radnika i ukupnu mesečnu platu svih radnika.

```
select count(*) "Broj radnika",
       sum(plt) "Ukupna mesecna plata"
from radnik;
```

- Šta je efekat sum funkcije kada u skupu ima null vrednosti?
  - Ignoriše null vrednosti

58

58

## Funkcija avg

- **AVG(<obeležje>)** – vraća srednju vrednost datog <obeležja>, za sve selektovane torke, uključujući višestruko ponavljanje istih torki
- **AVG(DISTINCT <obeležje>)** – vraća srednju vrednosti datog <obeležja>, za sve različite selektovane torke

59

59

## Funkcija avg

- Prikazati broj radnika, prosečnu platu i ukupnu godišnju platu svih radnika.  

```
select count(*) "Broj radnika",
       avg(plt) "Prosecna plata",
       12*sum(plt) "Godisnja plata"
from radnik;
```
- **Šta je efekat avg funkcije kada u skupu ima null vrednosti?**  
 – Ignoriše null vrednosti

60

60

## Skupovne funkcije nad isključivo null vrednostima

- Prikazati ukupnu premiju svih radnika čiji je matični broj veći od 100.  
**SELECT SUM(pre) FROM radnik WHERE mbr >100;**
- Šta je rezultat sum, avg, max, min funkcija kada su u skupu sve null vrednosti?  
Rezultat je null.
- Rezultat count funkcije u tom slučaju je vrednost 0.

61

61

## Funkcija round

- ROUND(<izraz>, <broj\_decimala>) – vraća zaokruženu vrednost datog <izraza> na dati <broj\_decimala>

62

62

## Funkcija round

- Prikazati prosečnu platu svih radnika pomnoženu sa koren iz 2 (1,41) zaokruženo na dve decimale.

```
select round(avg(plt *1.41), 2)  
from radnik;
```

63

63

## Select naredba u listi tabela

```
SELECT * FROM (SELECT MBR,IME  
FROM radnik);
```

64

64



## ROWNUM

Prikazati 10 radnika koji imaju najveću platu, sortiranih po plati u opadajućem redosledu

```
SELECT mbr, plt, rownum
FROM RADNIK WHERE ROWNUM <= 10
ORDER BY PLT DESC;
```

- Da li je ovo ispravno?

65

65

## ROWNUM

- Pseudokolona
  - ROWNUM vrednost koju torka dobija nastaje na osnovu redosleda kojim Oracle dobavlja torke
  - Vrednost za ROWNUM kreće od 1

```
SELECT mbr, plt, rownum FROM
(SELECT * FROM Radnik ORDER BY plt
desc)
WHERE ROWNUM <= 10;
```

66

66

## Zadatak avg, round – select naveden u listi kolona

- Prikazati za svakog radnika red koji sadrži njegovu platu, prosečnu platu i apsolutnu (ABS) razliku prosečne plate i njegove plate.

	PLT	PROSECNA_PLATA	RAZLIKA
1	10000	20690.91	10690.91
2	20000	20690.91	690.91
3	15000	20690.91	5690.91
4	8000	20690.91	12690.91
5	40000	20690.91	19309.09

```
SELECT PLT,
(SELECT ROUND(AVG(PLT), 2) FROM
radnik) as prosecna_plata,
ABS((SELECT ROUND(AVG(PLT), 2) from
RADNIK) – plt) as razlika FROM radnik;
```

67

67

## GROUP BY - Uvod

- SELECT mbr, spr FROM radproj where mbr < 40;

mbr	spr
10	10
20	20
10	30
30	30
30	40

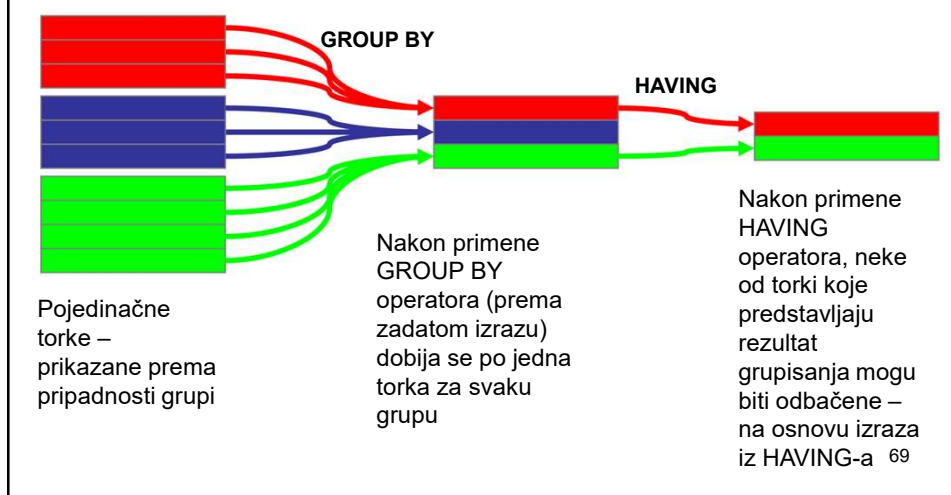
- SELECT mbr, count(spr) FROM radproj where mbr < 40 group by mbr;

mbr	count(spr)
30	2
20	1
10	2

68

68

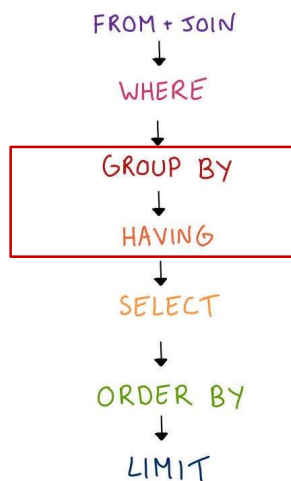
## GROUP BY - Uvod



69

## Redosled izvršavanja klauzula

Redosled izvršavanja SQL klauzula



70

70

## GROUP BY

- Prikazati koliko radnika radi na svakom projektu i koliko je ukupno angažovanje na tom projektu?

```
select spr, count(mbr), sum(brc)  
from radproj  
group by spr;
```

71

71

## HAVING

- Izlistati mbr radnika koji rade na više od dva projekta, pored mbr-a, prikazati i broj projekata na kojima radnici rade.

```
select mbr from radproj  
group by mbr  
having count(spr)>2;
```

```
select mbr, count(spr) from radproj  
group by mbr  
having count(spr)>2;
```

72

72

## GROUP BY - Napomene

- Najčešće se koristi u kombinaciji sa skupovnim funkcijama (min, max, count, avg...)
- Svaka kolona koja se nađe među izrazima SELECT klauzule, osim onih kolona koji su pod skupovnom funkcijom se mora naći i u izrazima GROUP BY klauzule
  - Npr count(spr) se može naći u izrazima u select klauzuli, a spr se ne mora naći naveden u izrazima koji pripadaju group by klauzuli
    - Ovakva upotreba i jeste najčešća
- Grupe se mogu filtrirati korišćenjem having ključne reči
  - Where filtrira torke, ne grupe
- Može se koristiti u kombinaciji sa ORDER BY

73

73

## Nezavisni ugnježdeni upiti

- SELECT unutar WHERE druge SELECT naredbe
- Predikatski izrazi:
  - ANY, ALL, IN i EXISTS
- SQL dozvoljava višestruko ugnježdavanje upita

74

74

## Nezavisni ugnježdeni upiti

- Izlistati u rastućem redosledu plate mbr, ime, prz i plt radnika koji imaju platu veću od prosečne.

```
select mbr, ime, prz, plt  
from radnik  
where plt>(select avg(plt) from radnik)  
order by plt asc;
```

75

75

## Nezavisni ugnježdeni upiti

- Izlistati imena i prezimena radnika koji rade na projektu sa šifrom 30

```
select ime, prz from radnik  
where mbr in  
(select mbr from radproj where spr=30);
```

76

76

## Nezavisni ugnježdjeni upiti

- Izlistati mbr, ime, prz radnika koji rade na projektu sa šifrom 10, a ne rade na projektu sa šifrom 30.

```
select mbr, ime, prz
from radnik
where mbr in
(select mbr from radproj where spr=10)
and mbr not in
(select mbr from radproj where spr=30);
```

- Zašto ne može u jednom ugnježđenom upitu?**

77

77

## Nezavisni ugnježdjeni upiti

- Izlistati ime, prz i god najstarijeg radnika.

```
select mbr, ime, prz, god
from radnik
where god <= all(select god from radnik);
```

```
select mbr, ime, prz, god
from radnik
where god = (select min(god) from
radnik);
```

78

78

## Spajanje tabela

- Prikazati mbr, prz, ime, plt i brc angažovanja svih radnika koji rade na projektu sa šifrom 10.

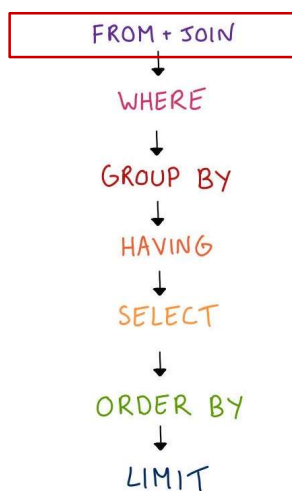
```
select radnik.mbr, prz, ime, plt, brc
from radnik, radproj
where spr=10 and
radnik.mbr = radproj.mbr;
```

79

79

## Redosled izvršavanja klauzula

Redosled izvršavanja SQL klauzula



80

80



## Spajanje tabela

- Prikazati mbr, prz, ime, plt i brc angažovanja svih radnika koji rade na projektu sa šifrom 10.

```
select r.mbr, r.prz, r.ime, r.plt, rp.brc  
from radnik r, radproj rp  
where rp.spr=10 and r.mbr=rp.mbr;
```

81

81

## Spajanje tabela

- Prikazati mbr, ime, prz i plt radnika koji su rukovodioci projekata.

```
select distinct mbr, ime, prz, plt  
from radnik, projekat  
where ruk=mbr;
```

82

82

## Spajanje tabela

- Izlistati imena, prezimena svih radnika osim rukovodioca projekta sa šifrom 10.

```
select mbr, ime, prz  
from radnik r, projekat p  
where p.spr=10 and r.mbr!=p.ruk;
```

83

83

## Primer

- Izlistati imena, prezimena svih radnika osim rukovodioca projekta sa šifrom 10. (sa ugnježenim upitom)

```
select ime, prz, mbr  
from radnik  
where mbr != (select ruk from projekat  
              where spr=10);
```

84

84

## Primer

- Izlistati nazive projekata na kojima radi bar jedan radnik koji radi i na projektu sa šifrom 60.

```
select p.nap from projekat p  
where spr in (select spr from radproj  
where mbr in (select mbr from radproj  
where spr=60));
```

85

85

## Primer

- Prikazati imena i prezimena rukovodilaca projekata i broj projekata kojima rukovode.

```
select prz, ime, count(spr)  
from radnik r, projekat p  
where ruk=mbr  
group by mbr, prz, ime;
```

86

86

## Primer

- Prikazati za svakog radnika mbr, prz, ime, ukupan broj projekata i ukupno angažovanje na projektima na kojima radi.

```
select r.mbr, r.prz, r.ime, count(*),  
sum(rp.brc)  
from radnik r, radproj rp  
where r.mbr=rp.mbr  
group by r.mbr, r.prz, r.ime;
```

87

87

## Primer

- Prikazati imena i prezimena rukovodilaca projekata i broj projekata na kojima **rade**.

```
select ime, prz, count(distinct rp.spr) bp from  
radnik r, radproj rp  
where r.mbr=rp.mbr and  
r.mbr in (select ruk from projekat)  
group by mbr, prz, ime;
```

```
select ime,prz,count(distinct rp.spr) from radnik  
r,projekat p, radproj rp where rp.mbr=r.mbr and  
p.ruk=r.mbr group by r.mbr,ime,prz;
```

88

88

## Primer

- Izlistati nazive projekata na kojima se ukupno radi više od 15 časova.

```
select nap  
from projekat p, radproj rp  
where p.spr=rp.spr  
group by p.spr, nap  
having sum(brc)>15;
```

89

89

## Primer

- Izlistati šifre i nazive projekata na kojima radi više od dva radnika.

```
select p.spr, p.nap  
from projekat p, radproj rp  
where rp.spr=p.spr  
group by p.spr, p.nap  
having count(mbr)>2;
```

90

90

## Primer

- Izlistati nazive i šifre projekata na kojima je prosečno angažovanje veće od prosečnog angažovanja na svim projektima.

```
select p.spr, p.nap
from projekat p, radproj rp
where rp.spr=p.spr
group by p.spr, p.nap
having avg(brc)>(select avg(brc) from radproj);
```

91

91

## Primer

- Izlistati nazive i šifre projekata sa najvećim prosečnim angažovanjem.

	SPR	MBR	BRC
1	10	10	10
2	10	50	11
3	10	100	9
4	10	130	2
5	20	20	13
6	20	70	8
7	20	110	8
8	20	120	7
9	30	10	4
10	30	30	3
11	30	50	5
12	30	60	4
13	30	80	9

Primer:

SPR = 10

$$(10+11+9+2)/4 = 8$$

SPR = 20

$$(13+8+8+7)/4 = 9$$

SPR = 30

$$(4+3+4+5+9)/5 = 5$$

Upit bi u ovom slučaju trebalo da vrati projekat sa šifrom 20

92

92

## Rešenje

```
select p.spr, p.nap
from projekat p, radproj rp
where rp.spr=p.spr
group by p.spr, p.nap
having avg(brc)>=all(select avg(brc)
from radproj group by spr);
```

93

93

## Upit sa višestrukom upotrebom iste tabele

- Prikazati mbr, ime, prz, plt radnika koji zarađuju više od radnika sa matičnim brojem 40.

```
select r.mbr, r.prz, r.ime, r.plt
from radnik r, radnik r1
where r.plt>r1.plt and r1.mbr=40;
```

94

94

## Upit sa višestrukom upotrebom iste tabele

- Prikazati imena, prezimena i plate radnika koji zarađuju bar 1000 dinara manje od rukovodioca projekta na kom radnik radi.

```
select r1.ime, r1.prz, r1.plt, p.nap from  
radnik r1, radnik r2, projekat p, radproj rp  
where r1.mbr=rp.mbr and rp.spr=p.spr and  
p.ruk=r2.mbr and r1.plt+1000<r2.plt;
```

95

95

## Povezani upiti

- Prikazati mbr, ime, prz, plt radnika čiji je broj sati angažovanja na nekom projektu veći od prosečnog broja sati angažovanja na tom projektu.

```
select distinct r.mbr, ime, prz, plt, brc  
from radnik r, radproj rp1  
where r.mbr=rp1.mbr and  
rp1.brc>(select avg(brc) from radproj rp2  
where rp2.spr=rp1.spr);
```

96

96



## EXISTS

**EXISTS(<lista\_vrednosti>)** –  
<lista\_vrednosti> nije prazan skup  
vrednosti

**NOT EXISTS(<lista\_vrednosti>)** –  
<lista\_vrednosti> je prazan skup vrednosti

97

97

## EXISTS

- Ko je najstariji radnik? (exist)

```
select ime, prz, god
from radnik r
where not exists
(select mbr from radnik r1
where r1.god<r.god);
```

98

98

## EXISTS

- Izlistati mbr, ime, prz radnika koji ne rade na projektu sa šifrom 10. (ne postoji radnik sa projekta 10 koji je jednak traženom radniku)

```
select mbr, ime, prz
from radnik r
where not exists
(select * from radproj rp
where r.mbr=rp.mbr and rp.spr=10);
```

99

99

## EXISTS

- Izlistati radnike koji ne rade ni na jednom projektu. (ne postoji projekat na kom rade)

```
select mbr, ime, prz
from radnik r
where not exists
(select * from radproj rp where r.mbr=rp.mbr);

select mbr, ime, prz
from radnik r
where mbr not in
(select rp.mbr from radproj rp
where r.mbr=rp.mbr);
```

100

100

## EXISTS

- Izlistati radnike koji nisu rukovodioci projekata. (ne postoji projekat kojim rukovodi taj radnik)

```
select mbr, ime, prz
from radnik r
where not exists
(select * from projekat where mbr=ruk);
```

```
select mbr, ime, prz
from radnik r
where mbr not in
(select ruk from projekat where mbr=ruk);
```

101

101

## Povezani upiti - EXISTS

- Ko je najmlađi rukovodilac projekata?

```
select distinct mbr, ime, prz, god
from radnik r, projekat p
where r.mbr=p.ruk and not exists
(select mbr from radnik r1, projekat p1
where r1.mbr=p1.ruk and r1.god>=r.god);
```

102

102

## Unija (UNION)

- Izlistati mbr, ime, prz radnika koji rade na projektu sa šifrom 20 ili im je plata veća od prosečne. (unija)

```
select mbr, ime, prz from radnik  
where mbr in  
(select mbr from radproj where spr=20)  
union  
select mbr, ime, prz from radnik  
where plt>(select avg(plt) from radnik);
```

103

103

## Unija (UNION ALL)

- Izlistati mbr, ime, prz radnika koji rade na projektu sa šifrom 20 ili im je plata veća od prosečne. (unija)

```
select mbr, ime, prz from radnik  
where mbr in  
(select mbr from radproj where spr=20)  
union all  
select mbr, ime, prz from radnik  
where plt>(select avg(plt) from radnik);
```

104

104

## Presek (INTERSECT)

- Izlistati mbr, ime, prz radnika čije prezime počinje na slovo M ili slovo R i mbr, ime, prz radnika čije prezime počinje na slovo M ili slovo P.

```
select mbr, ime, prz from radnik  
where prz like 'M%' or prz like 'R%'
```

**INTERSECT**

```
select mbr, ime, prz from radnik  
where prz like 'M%' or prz like 'P%';
```

105

105

## Razlika (MINUS)

- Izlistati mbr, ime, prz radnika čije prezime počinje na slovo M ili slovo R i mbr, ime, prz radnika čije prezime počinje na slovo M ili slovo P.

```
select mbr, ime, prz from radnik  
where prz like 'M%' or prz like 'R%'
```

**MINUS**

```
select mbr, ime, prz from radnik  
where prz like 'M%' or prz like 'P%';
```

106

106

## Prirodno spajanje (NATURAL)

- Prikazati ime i prz radnika koji rade na projektu sa šifrom 30.

```
select ime, prz  
from radnik natural join radproj  
where spr=30;
```

Spajanje se vrši na osnovu imena kolona.

107

107

## Unutrašnje spajanje (INNER)

- Prikazati ime i prz radnika koji rade na projektu sa šifrom 30.

```
select ime, prz  
from radnik r inner join radproj rp  
on r.mbr=rp.mbr  
where spr=30;
```

108

108

## Spoljno spajanje (OUTER)

- Levo (LEFT)
- Desno (RIGHT)
- Potpuno (FULL)

109

109

## Spoljno spajanje (LEFT OUTER)

- Prikazati mbr, ime i prz radnika i šifre projekata na kojima rade. Prikazati, takođe, iste podatke i za radnike koji ne rade ni na jednom projektu, pri čemu za šifru projekta treba, u tom slučaju, prikazati nedostajuću vrednost.

```
select r.mbr,ime, prz, spr  
from radnik r left outer join radproj rp  
on r.mbr=rp.mbr;
```

110

110

## Spoljno spajanje (LEFT OUTER)

- Prikazati mbr, ime i prz svih radnika i nazive projekata kojima rukovode. Ukoliko radnik ne rukovodi ni jednim projektom ispisati: ne rukovodi projektom.

```
select r.mbr,ime, prz, nvl(nap, 'ne rukovodi
projektom') Projekat
from radnik r left outer join projekat p
on r.mbr=p.ruk;
```

111

111

## Spoljno spajanje (RIGHT OUTER)

- Prikazati nazive svih projekata i mbr radnika koji rade na njima. Ukoliko na projektu ne radi ni jedan radnik ispisati nulu umesto matičnog broja.

```
select nvl(rp.mbr, 0) "Mbr radnika", nap
from radproj rp right outer join projekat p
on rp.spr=p.spr;
```

```
select nvl(rp.mbr, 0) "Mbr radnika", nap
from radproj rp, projekat p
where rp.spr(+) = p.spr;
```

112

112



## Spoljno spajanje (FULL OUTER)

```
select nvl(rp.mbr, 0) "Mbr radnika", nap  
from radproj rp full outer join projekat p  
on rp.spr=p.spr;
```

113

113

## Primer

- Prikazati matične brojeve, imena i prezimena radnika, zajedno sa šiframa projekata na kojima rade. Prikazati, takođe, iste podatke i za radnike koji ne rade ni na jednom projektu, pri čemu za šifru projekta treba, u tom slučaju, prikazati nedostajuću vrednost.

114

114

## Rešenje

```
SELECT r.Mbr, r.Prz, r.Ime, rp.Spr  
FROM Radnik r, Radproj rp  
WHERE r.Mbr = rp.Mbr (+);
```

```
SELECT r.Mbr, r.Prz, r.Ime, rp.Spr  
FROM Radnik r LEFT OUTER JOIN  
Radproj rp ON r.Mbr = rp.Mbr;
```

115

115