



Računarstvo i informatika

Katedra za računarstvo

Elektronski fakultet u Nišu

Baze podataka (Računske vežbe) **SQL SELECT naredba (I)**

Letnji semestar 2016/2017



Sadržaj

- Naredba SELECT
- Klauzule SELECT i FROM
- Klauzula WHERE
- Klauzula ORDER BY
- Aritmetičke funkcije
- Funkcije za rad sa tekstualnim podacima
- Funkcije agregacije



Naredba SELECT

- Pretraživanje i pribavljanje podataka su najčešće operacije koje korisnici izvršavaju u relacionoj bazi podataka.
- Za pretraživanje i pribavljanje podataka SQL programski jezik obezbeđuje naredbu SELECT.
- Naredba SELECT pribavlja podatke iz jedne tabele ili više povezanih tabela koje se nalaze u relacionoj bazi podataka.
- U svom osnovnom obliku naredba SELECT ne može ni na koji način da izmeni podatke koji se nalaze u relacionoj bazi podataka.



Naredba SELECT

- Naredba SELECT je deklarativna naredba. Korišćenjem ove naredbe korisnici imaju mogućost samo da specificiraju rezultate koje žele.
- Sa druge strane RDBMS je zadužen da isplanira, optimizuje i izvrši fizičke operacije neophodne za generisanje specificiranih rezultata.
- Rezultat SELECT naredbe je uvek relacija.
- Naredba SELECT koristi podatke iz jedne ili većeg broja tabela, manipulište tim podacima i kao rezultat generiše tabelu.
- Kada je rezultat obrade skalarna vrednost ona se tretira kao tabela sa jednom vrstom i jednom kolonom.



Naredba SELECT

- Naredba SELECT je jedna od najkompleksnijih naredbi SQL jezika i uključuje veliki broj ključnih reči klauzula:
 - **SELECT** – definiše listu kolona koja će biti uključena u rezultujuću tabelu.
 - **FROM** – definiše tabele iz kojih se pribavljaju podaci za potrebe generisanja rezultujuće tabele. Može da uključi jednu ili više JOIN klauzula za povezivanje tabela na osnovu kriterijuma zadatih od strane korisnika.
 - **WHERE** definiše predikat na osnovu koga se ograničava broj vrsta u rezultujućoj tabeli. Ova klauzula iz rezultata eliminiše sve vrste za koje specificirani predikat ne vraća vrednost TRUE.
 - **CONNECT BY** i **START WITH** – transformišu podatke u strukturu tipa stabla
 - **GROUP BY** – grupiše vrste u rezultujućoj tabeli koje u specificiranim kolonama imaju istu vrednost.
 - **HAVING** – definiše predikat na osnovu koga se eliminišu vrste nakon što je klauzula GROUP BY primenjena na rezultujuću tabelu.
 - **ORDER BY** – koristi se za sortiranje rezultujuće tabele. Korisnici definišu kolone po kojima se vrši sortiranje kao i smer sortiranja.
 - **UNION, INTERSECT, MINUS** – primenjuju se na rezultujućim tabelama većeg broja SQL naredbi.



Klauzule SELECT i FROM

- Klauzule SELECT i FROM su jedine obavezne u okviru SELECT naredbe.
- Klauzula FROM specificira tabele iz kojih se pribavljaju podaci.
- Ukoliko se navede više tabela potrebno je specificirati uslove spajanja tabela.
- Inicijalno razmatramo samo na pribavljanje podataka iz jedne tabele.



Klauzule SELECT i FROM

- Klauzula SELECT specificira kolone koje treba uključiti u rezultujuću tabelu. Mogu se koristiti sledeće opcije:
 - **ALL** – u rezultujućoj tabeli prikazuju se sve vrste koje zadovoljavaju navedeni predikat
 - **DISTINCT** – iz rezultujuće tabele izbacuju se duplikati vrsta
 - ***** - rezultujuća tabela uključuje sve kolone tabele ili tabela iz kojih se pribavljaju podaci
 - **tabela.*** - rezultujuća tabela uključuje sve kolone specificirane tabele
 - **izraz** - ime kolone ili funkcije nad kolonama koja će biti uključena u rezultujuću tabelu
 - **AS pseudonim** - novo ime kolone ili funkcije nad kolonama koje im se dodeljuje u rezultujućoj tabeli



Klauzule SELECT i FROM

- SELECT naredba koja prikazuje kompletan sadržaj tabele RADNIK.

SELECT * **FROM** RADNIK;

<small>RZ</small> LIME	<small>RZ</small> SSLOVO	<small>RZ</small> PREZIME	<small>RZ</small> MATBR	<small>RZ</small> DATRODJ	<small>RZ</small> POL	<small>RZ</small> PLATA	<small>RZ</small> ADRESA	<small>RZ</small> MATBRS	<small>RZ</small> BRSEK
Marko	J	Petrović	123456789	09-JAN-65	M	30000	Obilićev Venac 11	333445555	5
Sima	F	Todorović	333445555	08-DEC-55	M	40000	Dušanova 32	888665555	5
Valentina	D	Kovačević	999887777	19-JAN-68	Ž	25000	Knjeginje Ljubice 12/34	987654321	4
Aleksandra	S	Petrović	987654321	20-JUN-41	Ž	43000	Knjaževačka 11	888665555	4
Velibor	T	Jovanović	666884444	15-SEP-62	M	36000	Knjaževačka 132/12	333445555	5
Jelena	P	Janković	453453453	31-JUL-72	Ž	25000	Vizantijski bulevar 123/12	333445555	5
Stanko	L	Manojlović	987987987	29-MAR-69	M	25000	Nemanjina 23	987654321	4
Jovan	S	Obradović	888665555	10-NOV-47	M	55000	Nikole Kopernika 11	(null)	1

- Rezultat bi bio ekvivalenta da smo napisali upit kod koga su umesto * navedena imena svih kolona u tabeli.

SELECT LIME, SSLOVO, PREZIME, MATBR, DATRODJ,
POL, PLATA, MATBRS, BRSEK
FROM RADNIK;



Klauzule SELECT i FROM

- Ukoliko želimo da prikazemo samo određene kolone iz tabele RADNIK posle SELECT klauzule navešćemo imena kolona koje su od interesa. U nastavku je dat SQL upit koji prikazuje samo imena i prezimena radnika.

SELECT IME, PREZIME **FROM** RADNIK;



IME	PREZIME
Marko	Petrović
Sima	Todorović
Valentina	Kovačević
Aleksandra	Petrović
Velibor	Jovanović
Jelena	Janković
Stanko	Manojlović
Jovan	Obradović



Klauzule SELECT i FROM

- Redosled kojim su kolone navedene u klauzuli SELECT definiše redosled kolona u rezultujućoj tabeli. U nastavku je dat SQL upit koji prikazuje imena i prezimena svih radnika ali u nešto drugačijem redosledu.

SELECT PREZIME, LIME **FROM** RADNIK;

 PREZIME	 LIME
Petrović	Marko
Todorović	Sima
Kovačević	Valentina
Petrović	Aleksandra
Jovanović	Velibor
Janković	Jelena
Manojlović	Stanko
Obradović	Jovan



Klauzule SELECT i FROM

- U nastavku je dat SQL upit koji za svakog radnik određuje matični broj njegovog neposrednog rukovodioca.

SELECT MATBRS **FROM** RADNIK;

MATBRS
333445555
888665555
987654321
888665555
333445555
333445555
987654321
(null)

- Ekvivalentan upit koristi ključnu reč ALL. Prikazuje sve vrste rezultujuće tabe (uključujući i duplikate).
- Ključna reč ALL je podrazumevana pa se najčešće izostavlja.

SELECT ALL MATBRS **FROM** RADNIK;



Klauzule SELECT i FROM

- U prethodnom primeru u rezultujućoj tabeli neki matični brojevi su se pojavili više puta.
- To je posledica činjenice da veći broj radnika može imati istog rukovodioca.
- Za eliminisanje duplikata iz rezultujuće tabele koristi se ključna reč DISTINCT.

SELECT DISTINCT MATBRS **FROM** RADNIK;

MATBRS
333445555
(null)
987654321
888665555



Klauzula WHERE

- Klauzula WHERE specificira uslov na osnovu koga se kreira rezultujuća tabela.
- U rezultujuću tabelu će biti uključene samo one vrste koje zadovoljavaju specificirani uslov odnosno vrste za koje specificirani uslov ima vrednost TRUE.
- U uslovu se mogu javiti:
 1. Relacioni operatori
 2. Logički operatori
 3. Operator BETWEEN
 4. Operator IN
 5. Operator LIKE
 6. Operator IS NULL



Klauzula WHERE

- SQL podržava šest relacionih operatora koji imaju sledeće značenje:

1. = Jednako
2. <> Nije jednako (različito)
3. < Manje od
4. > Veće od
5. <= Manje ili jednako a
6. >= Veće ili jednako

```
SELECT *  
FROM RADNIK  
WHERE PREZIME = 'Petrović';
```

SQL upit prikazuje sve podatke o radnicima koje se prezivaju Petrović.

Tekstualni podaci se zadaju korišćenjem jednostrukih znakova navoda

IME	SSLOVO	PREZIME	MATBR	DATRODJ	POL	PLATA	ADRESA	MATBR5	BRSEK
Marko	J	Petrović	123456789	09-JAN-65	M	30000	Obilićev Venac 11	333445555	5
Aleksandra	S	Petrović	987654321	20-JUN-41	Ž	43000	Knjaževačka 11	888665555	4





Klauzula WHERE

- SQL upit koji prikazuje imena i prezimena radnika čija je plata jednaka ili veća od 40000.

SELECT LIME, PREZIME

FROM RADNIK

WHERE PLATA >= 40000;

 LIME	 PREZIME
Sima	Todorović
Aleksandra	Petrović
Jovan	Obradović



Klauzula WHERE

- SQL omogućava korišćenje standardnih logičkih operatore AND, OR i NOT, ali i operatore IN i BETWEEN koji omogućavaju jednostavnije korišćenje prethodno navedenih operatora u nekim slučajevima.
- Prioritet logičkih operatora je sledeći:
 1. NOT
 2. AND
 3. OR
- Logički operator negacije NOT se navodi na početku logičkog izraza, a ne ispred operatora poređenja. **Na primer, NOT A = B je validni WHERE uslov, ali A NOT = B nije.**





Klauzula WHERE

- SQL upit koji prikazuje podatke o radnicima koji se prezivaju Petrović i čija je plata jednaka ili veća od 40000.

SELECT LIME, PREZIME

FROM RADNIK

WHERE PREZIME = 'Petrović' AND PLATA >= 40000;

 LIME	 PREZIME
Aleksandra	Petrović



Klauzula WHERE

- SQL upit koji prikazuje podatke o radnicima koji se prezivaju Petrović i čija je plata manja od 40000 (nije jednaka ili veća od 40000).

SELECT LIME, PREZIME

FROM RADNIK

WHERE PREZIME = 'Petrović' AND NOT PLATA >= 40000;

AZ	LIME	AZ	PREZIME
	Marko		Petrović





Klauzula WHERE

- SQL upit koji prikazuje podatke o radnicima koji se prezivaju Petrović ili se prezivaju Jovanović.

SELECT LIME, PREZIME

FROM RADNIK

WHERE PREZIME = 'Petrović' **OR** PREZIME = 'Jovanović';

 LIME	 PREZIME
Marko	Petrović
Aleksandra	Petrović
Velibor	Jovanović



Klauzula WHERE

- Operator IN zamenjuje višestruku upotrebu operatora OR i =.
- Operator NOT IN prikazuje sve vrste osim onih određenih IN listom.
- SQL upit koji korišćenjem operatora IN izdavaća samo radnike koji se prezivaju Jovanović ili Petrović.

SELECT LIME, PREZIME

FROM RADNIK

WHERE PREZIME IN ('Petrović', 'Jovanović');

AZ LIME	AZ PREZIME
Marko	Petrović
Aleksandra	Petrović
Velibor	Jovanović




Klauzula WHERE

- SQL upit koji pribavlja podatke o svim rancima koji se ne prezivaju Petrović ili Jovanović.

SELECT LIME, PREZIME

FROM RADNIK

WHERE PREZIME **NOT IN** ('Petrović', 'Jovanović');

 LIME	 PREZIME
Sima	Todorović
Valentina	Kovačević
Jelena	Janković
Stanko	Manojlović
Jovan	Obradović



Klauzula WHERE

- Operator BETWEEN zamenjuje višestruku upotrebu operatora AND i =. Ovaj operator omogućava ispitivanje da li je vrednost atributa/kolone u zadatom opsegu.
- SQL upit koji prikazuje podataka o radnicima čija je plata u opsegu od 30000 do 40000 (uključujući i granice opsega).

SELECT *

FROM RADNIK

WHERE PLATA BETWEEN 30000 AND 40000;

LIME	SSLOVO	PREZIME	MATBR	DATRODJ	POL	PLATA	ADRESA	MATBRS	BRSEK
Marko	J	Petrović	123456789	09-JAN-65	M	30000	Obilićev Venac 11	333445555	5
Sima	F	Todorović	333445555	08-DEC-55	M	40000	Dušanova 32	888665555	5
Velibor	T	Jovanović	666884444	15-SEP-62	M	36000	Knjaževačka 132/12	333445555	5



Klauzula WHERE

- Operator IS NULL se koristi za poređenje sa NULL vrednostima (da li je vrednost kolone NULL).
- **Treba voditi računa da se na NULL vrednosti ne može primeniti ni jedan relacioni operator (sve NULL vrednosti su jedinstvene).**
- Može se samo proveravati da li kolona ima NULL vrednost ili nema.
- SQL upit koji pronalazi radnika koji nema neposrednog rukovodioca.

SELECT *

FROM RADNIK

WHERE MATBRS IS NULL;

IME	SSLOVO	PREZIME	MATBR	DATRODJ	POL	PLATA	ADRESA	MATBRS	BRSEK
Jovan	S	Obradović	888665555	10-NOV-47	M	55000	Nikole Kopernika 11	(null)	1



Klauzula WHERE

- Naredni SQL upit ne vraća ni jedan slog u rezultujućoj tabeli jer poređenje sa NULL vrednošću nije moguće. Oracle ne prijavljuje grešku.

SELECT *

FROM RADNIK

WHERE MATBRS = NULL;

- Negacija poređenja IS NULL.

SELECT *

FROM RADNIK

WHERE NOT MATBRS IS NULL;

SELECT *

FROM RADNIK

WHERE MATBRS IS NOT NULL;



Klauzula WHERE

- Operator LIKE omogućava poređenje znakovnih vrednosti (tekstualnih podataka) sa zadatim šablonom.
- Za definisanje šablona koriste se simboli procenat (%) i donja crta (_).
- Procenat (%) predstavlja bilo koji znak (broj, slovo, interpunkcijski znak) ili skup znakova. Procenat (%) može da zameni i prazan string.
- Donja crta (_) zamenjuje samo jedan znak (Pojavljivanje znaka je obavezno).
- Operator NOT LIKE prikazuje sve vrste koje ne odgovaraju prethodno zadatom opisu



Klauzula WHERE

- SQL upit koji prikazuje sve radnike koji imaju slovo J n apočetku prezimena.

SELECT *

FROM RADNIK

WHERE PREZIME LIKE 'J%';

IME	SSLOVO	PREZIME	MATBR	DATRODJ	POL	PLATA	ADRESA	MATBR5	BRSEK
Velibor	T	Jovanović	666884444	15-SEP-62	M	36000	Knjaževačka 132/12	333445555	5
Jelena	P	Janković	453453453	31-JUL-72	Ž	25000	Vizantijski bulevar 123/12	333445555	5

- **Oracle kod tekstualnih podataka pravi razliku između malih i velikih slova.**
- Predikat LIKE 'j%' ne bi vratio nijedan slog.



Klauzula WHERE

- SQL upit koji prikazuje sve radnike rođene septembra.

SELECT *

FROM RADNIK

WHERE DATRODJ LIKE '%SEP%';

- Alternativni upit:

SELECT *

FROM RADNIK

WHERE DATRODJ LIKE '____SEP____';

IME	SSLOVO	PREZIME	MATBR	DATRODJ	POL	PLATA	ADRESA	MATBR5	BRSEK
Velibor T		Jovanović	666884444	15-SEP-62	M	36000	Knjaževačka 132/12	333445555	5



Klauzula WHERE

- SQL upit koji prikazuje sve radnike koji nisu rođeni septembra.

SELECT *

FROM RADNIK

WHERE DATRODJ NOT LIKE '____SEP____';

IME	SSLOVO	PREZIME	MATBR	DATRODJ	POL	PLATA	ADRESA	MATBRS	BRSEK
Marko	J	Petrović	123456789	09-JAN-65	M	30000	Obilićev Venac 11	333445555	5
Sima	F	Todorović	333445555	08-DEC-55	M	40000	Dušanova 32	888665555	5
Valentina	D	Kovačević	999887777	19-JAN-68	Ž	25000	Knjeginje Ljubice 12/34	987654321	4
Aleksandra	S	Petrov_ić	987654321	20-JUN-41	Ž	43000	Knjaževačka 11	888665555	4
Jelena	P	Janković	453453453	31-JUL-72	Ž	25000	Vizantijski bulevar 123/12	333445555	5
Stanko	L	Manojlović	987987987	29-MAR-69	M	25000	Nemanjina 23	987654321	4
Jovan	S	Obradović	888665555	10-NOV-47	M	55000	Nikole Kopernika 11	(null)	1



Klauzula WHERE

- Korišćenje % i _ unutar šablona.

SELECT *

FROM RADNIK

WHERE PREZIME LIKE 'Pe%_%' ESCAPE '\';

A2	LIME	A2	SSLOVO	A2	PREZIME	A2	MATBR	A2	DATRODJ	A2	POL	A2	PLATA	A2	ADRESA	A2	MATBRS	A2	BRSEK
	Aleksandra		S		Petrov_ić		987654321		20-JUN-41		Ž		43000		Knjaževačka 11		888665555		4



Klauzula ORDER BY

- Klauzula ORDER BY specificira redosled prikazivanja vrsta rezultujuće tabele sortiranjem po vrednosti nekih kolona u rastući (ASC) ili opadajući redosled (DESC).
- Ukoliko klauzula ORDER BY nije navedena vrste u rezultujućoj tabeli su poređane po slučajnom principu i ne postoji nikakva garancija da će isti upit uvek generisati rezultujuću tabelu čije su vrste poređane na isti način.



Klauzula ORDER BY

- SQL upit koji prikazuje podatke o radnicima i sortira ih prema prezimenu u rastućem redosledu.
- Smer sortiranja ASC je podrazumevan i može da se izostavi.

SELECT *

FROM RADNIK

ORDER BY PREZIME ASC;

RZ	LIME	RZ	SSLOVO	RZ	PREZIME	RZ	MATBR	RZ	DATRODJ	RZ	POL	RZ	PLATA	RZ	ADRESA	RZ	MATBR	RZ	BRSEK
	Jelena		P		Janković		453453453		31-JUL-72		Ž		25000		Vizantijski bulevar 123/12		333445555		5
	Velibor		T		Jovanović		666884444		15-SEP-62		M		36000		Knjaževačka 132/12		333445555		5
	Valentina		D		Kovačević		999887777		19-JAN-68		Ž		25000		Knjeginje Ljubice 12/34		987654321		4
	Stanko		L		Manojlović		987987987		29-MAR-69		M		25000		Nemanjina 23		987654321		4
	Jovan		S		Obradović		888665555		10-NOV-47		M		55000		Nikole Kopernika 11		(null)		1
	Marko		J		Petrović		123456789		09-JAN-65		M		30000		Obilićev Venac 11		333445555		5
	Aleksandra		S		Petrović		987654321		20-JUN-41		Ž		43000		Knjaževačka 11		888665555		4
	Sima		F		Todorović		333445555		08-DEC-55		M		40000		Dušanova 32		888665555		5



Klauzula ORDER BY

- SQL upit koji prikazuje podatke o radnicima i sortira ih prema prezimenu u opadajućem redosledu.
- Opadajući redosled se uvek mora eksplicitno navesti.

SELECT *

FROM RADNIK

ORDER BY PREZIME DESC;

IME	SSLOVO	PREZIME	MATBR	DATRODJ	POL	PLATA	ADRESA	MATBR5	BRSEK
Sima	F	Todorović	333445555	08-DEC-55	M	40000	Dušanova 32	888665555	5
Marko	J	Petrović	123456789	09-JAN-65	M	30000	Obilićev Venac 11	333445555	5
Aleksandra	S	Petrović	987654321	20-JUN-41	Ž	43000	Knjaževačka 11	888665555	4
Jovan	S	Obradović	888665555	10-NOV-47	M	55000	Nikole Kopernika 11	(null)	1
Stanko	L	Manojlović	987987987	29-MAR-69	M	25000	Nemanjina 23	987654321	4
Valentina	D	Kovačević	999887777	19-JAN-68	Ž	25000	Knjeginje Ljubice 12/34	987654321	4
Velibor	T	Jovanović	666884444	15-SEP-62	M	36000	Knjaževačka 132/12	333445555	5
Jelena	P	Janković	453453453	31-JUL-72	Ž	25000	Vizantijski bulevar 123/12	333445555	5



Klauzula ORDER BY

- Sortiranje je moguće vršiti na osnovu vrednosti više kolona.
- SQL upit sortira podatke o radnicima prema broju sektora u opadajućem redosledu a prema prezimenu u rastućem redosledu.

SELECT *

FROM RADNIK

ORDER BY BRSEK DESC, PREZIME ASC;

RZ	LIME	RZ	SSLOVO	RZ	PREZIME	RZ	MATBR	RZ	DATRODJ	RZ	POL	RZ	PLATA	RZ	ADRESA	RZ	MATBRS	RZ	BRSEK
	Jelena		P		Janković		453453453		31-JUL-72		Ž		25000		Vizantijski bulevar 123/12		333445555		5
	Velibor		T		Jovanović		666884444		15-SEP-62		M		36000		Knjaževačka 132/12		333445555		5
	Marko		J		Petrović		123456789		09-JAN-65		M		30000		Obilićev Venac 11		333445555		5
	Sima		F		Todorović		333445555		08-DEC-55		M		40000		Dušanova 32		888665555		5
	Valentina		D		Kovačević		999887777		19-JAN-68		Ž		25000		Knjeginje Ljubice 12/34		987654321		4
	Stanko		L		Manojlović		987987987		29-MAR-69		M		25000		Nemanjina 23		987654321		4
	Aleksandra		S		Petrović		987654321		20-JUN-41		Ž		43000		Knjaževačka 11		888665555		4
	Jovan		S		Obradović		888665555		10-NOV-47		M		55000		Nikole Kopernika 11		(null)		1



Aritmetičke funkcije

- SQL dozvoljava korišćenje matematičkih funkcija u SELECT i WHERE klauzulama. Na taj način se kao rezultat pretraživanja mogu prikazati rezultati izračunavanja nekog matematičkog izraza.
 - +, -, *, /
 - ROUND(broj[,d]) - zaokruživanje na d decimala
 - POWER(broj,e) - e-ti stepen zadatog broja
 - TRUNC(broj[,d]) - odsecanje na d decimala
 - ABS(broj) - apsolutna vrednost broja
 - SIGN(broj) - znak broja
 - MOD(broj1, broj2) - broj1 mod broj2
 - SQRT(broj) - kvadratni koren broja
 - LEAST(izraz, ...) - najmanji navedeni izraz
 - GREATEST(izraz, ...) - najveći navedeni izraz



Aritmetičke funkcije

- SQL upit koji prikazuje imena i prezimena radnika kao i njihove plate uvećane za bonus od 5000.
- Rezultatu matematičke funkcije (Plata + 5000) dodeljeno je ime korišćenjem sintakse pseudonima:

(Plata + 5000) **PlataSaBonusom**

```
SELECT LIME, PREZIME,  
        PLATA + 5000 AS PLATA_SA_BONUSOM  
FROM RADNIK;
```

LIME	PREZIME	PLATA_SA_BONUSOM
Marko	Petrović	35000
Sima	Todorović	45000
Valentina	Kovačević	30000
Aleksandra	Petrović	48000
Velibor	Jovanović	41000
Jelena	Janković	30000
Stanko	Manojlović	30000
Jovan	Obradović	60000



Aritmetičke funkcije

- Aritmetičke funkcije se mogu primenjivati i u WHERE klauzuli.

```
SELECT LIME, PREZIME,
```

```
        PLATA + 5000 AS PLATA_SA_BONUSOM
```

```
FROM RADNIK
```

```
WHERE PLATA + 5000 > 40000;
```

- Pseudonim se može primeniti na bilo koju kolonu u SELECT klauzuli.

```
SELECT LIME, PREZIME, PLATA + 5000 AS PLATA_SA_BONUSOM,
```

```
       I AS KONSTANTA, NULL NOVA_KOLONA
```

```
FROM RADNIK;
```

R	LIME	R	PREZIME	R	PLATA_SA_BONUSOM	R	KONSTANTA	R	NOVA_KOLONA
	Marko		Petrović		35000		1 (null)		
	Sima		Todorović		45000		1 (null)		
	Valentina		Kovačević		30000		1 (null)		
	Aleksandra		Petrović		48000		1 (null)		
	Velibor		Jovanović		41000		1 (null)		
	Jelena		Janković		30000		1 (null)		
	Stanko		Manojlović		30000		1 (null)		
	Jovan		Obradović		60000		1 (null)		



Aritmetičke funkcije

- SQL upit koji određuje zaradu po satu svih radnika, zaokruženu na dve decimale. U mesecu postoji 22 radna dana, 8 sati po danu.

```
SELECT IME, PREZIME,  
       ROUND(PLATA/(22*8), 2) ZARADA_SAT  
FROM RADNIK;
```

IME	PREZIME	ZARADA_SAT
Marko	Petrović	170.45
Sima	Todorović	227.27
Valentina	Kovačević	142.05
Aleksandra	Petrović	244.32
Velibor	Jovanović	204.55
Jelena	Janković	142.05
Stanko	Manojlović	142.05
Jovan	Obradović	312.5

Funkcije za rad sa tekstualnim podacima



- Funkcije za rad sa tekstualnim podacima:
 - string1 || string2 - konkatencija stringova
 - LENGTH(string) - dužina stringa
 - SUBSTR(s, i, j) - podniz od s dužine j od pozicije i
 - INSTR(s1, s2[,k]) - traži s2 u s1 od pozicije k
 - UPPER(s) - prevodi s u velika slova
 - LOWER(s) - prevodi s u mala slova
 - TO_NUM(s) - prevodi s u numerički tip
 - TO_CHAR(n) - prevodi numerički u znakovni tip
 - LPAD(s, l[,c]) - popunjava s sa leve strane sa l znakova c
 - RPAD(s, l[,c]) - popunjava s sa desne strane sa l znakova c
 - NVL(s1, s2) - ako važi s1 IS NULL, vraća s2; u suprotnom vraća s1.

Funkcije za rad sa tekstualnim podacima



- SQL upit koji prikazuje informacije o radnicima imatične brojeve njihovih šefova. Za radnika koji nema neposrednog rukovodioca ispisuje se poruka NEMA ŠEFA.

```
SELECT LIME, PREZIME,  
       NVL(TO_CHAR(MATBRS), 'NEMA ŠEFA') SEF  
FROM RADNIK;
```

LIME	PREZIME	SEF
Marko	Petrović	333445555
Sima	Todorović	888665555
Valentina	Kovačević	987654321
Aleksandra	Petrović	888665555
Velibor	Jovanović	333445555
Jelena	Janković	333445555
Stanko	Manojlović	987654321
Jovan	Obradović	NEMA ŠEFA

Funkcije za rad sa tekstualnim podacima



- SQL upit koji korišćenjem konkatencije tekstualnih podataka kreira i prikazuje puno ime svih radnika.

```
SELECT LIME || ' ' || SSLOVO || '.' || ' ' || PREZIME AS PUNO_IME  
FROM RADNIK;
```

PUNO_IME
Marko J. Petrović
Sima F. Todorović
Valentina D. Kovačević
Aleksandra S. Petrović
Velibor T. Jovanović
Jelena P. Janković
Stanko L. Manojlović
Jovan S. Obradović



Funkcije agregacije

- Funkcije agregacije su dobile naziv po tome što vrše agregaciju rezultata upita.
- Korišćenje ovih funkcija je jednostavno, pošto se navode u listi kolona SELECT klauzule koje se prikazuju.
- Značenje funkcija je sledeće:
 - AVG(kolona) - izračunava srednju vrednost datog atributa
 - SUM(kolona) - izračunava sumu svih vrednosti atributa
 - MIN(kolona) - nalazi minimalnu vrednost atributa
 - MAX(kolona) - nalazi najveću vrednost atributa
 - COUNT(*) - nalazi broj vrsta u tabeli (grupi)
 - COUNT(kolona) - nalazi broj broj vrsta sa ne NULL vrednostima kolone
 - COUNT (DISTINCT kolona) - nalazi broj vrsta sa različitim vrednostima zadate kolone



Funkcije agregacije

- SQL naredba koja prikazuje broj vrsta u rezultujućoj tabeli.

```
SELECT COUNT(*) AS UKUPNO_RADNIKA  
FROM RADNIK;
```

UKUPNO_RADNIKA
8



Funkcije agregacije

- SQL upit koji određuje maksimalnu, minimalnu, prosečnu i ukupnu platu svih radnika.

```
SELECT MAX(PLATA) AS MAX_PLATA,  
       MIN(PLATA) AS MIN_PLATA,  
       AVG(PLATA), SUM(PLATA)  
FROM RADNIK;
```

MAX_PLATA	MIN_PLATA	AVG(PLATA)	SUM(PLATA)
55000	25000	34875	279000



Funkcije agregacije

- Funkcije agregacije nije moguće koristiti u WHERE klauzuli.
- To je posledica činjenice da se rezultat funkcija agregacija izračunava nakon što se odrede vrste koje ulaze u sastav rezultujuće tabele, odnosno nakon obrade predikta koji je zadat u WHERE klauzuli.
- **U nastavku je dat SQL upit koji se NE MOŽE IZVRŠITI i koji će GENERISATI GREŠKU.**

```
SELECT LIME, PREZIME, PLATA  
FROM RADNIK  
WHERE Plata > AVG(Plata).
```