

Jezik Baze Podataka SQL

SQL

- Jezik koji se koristi u radu sa relacionim bazama podataka
- Nije programski jezik i manje je kompleksan.
- Koristi se isključivo u radu za bazama podataka.
- SQL nije case sensitive, nije osetljiv na razmake.
- SQL upit se završava (;)

- “S.Q.L.” ili “sequel”
- Podržan od strane svih komercijalnih baza podataka
- Standardizovan – ogromna dokumentacija
- Unos i izvršenje SQL komandi putem grafičkog interfejsa, prompta ili je implementiran u program
- Deklerativni jezik, zasnovan na relacionoj algebri

Data Definition Language (DDL)

- Kreiranje i brisanje tabela
- Kreiranje i brisanje indeksa
- Kreiranje i brisanje pogleda

Data Manipulation Language (DML)

- Select
- Insert
- Update
- Delete

Ostale komande

indeksi, ograničenja, pogledi, trigeri, transakcije, autorizacija, ...

Osnovni SELECT iskaz

select A_1, A_2, \dots, A_n

Atributi koji će se prikazati

From R_1, R_2, \dots, R_m

Relacije koje su uključene

where uslov

Filtriranje tj. kombinovanje redova na osnovu uslova

ISKAZ PRIMENOM OPERACIJE RELACIONE ALGEBRE

$$\pi_{A_1, A_2, \dots, A_n}(\sigma_{uslov}(R_1 \times R_2 \times \dots \times R_n))$$

SQL – Select (Čitanje podataka)

```
SELECT columns  
FROM table
```

Klijent
KlijentID (PK)
Ime
Prezime
Email
Telefon
Adresa1
Dobit

```
SELECT Email  
FROM Klijent
```

Email
milan.s@vtsnis.edu.rs
ana.m@gmail.com
ena.m@yahoo.com
darko.n@itc.rs
bojan.l@vrs.com

SQL – IDENTIFIKACIJA BAZE

```
SELECT *  
FROM E_Komerc.Klijent
```



Naziv Baze

SQL – WHERE (Filtriranje podataka)

- Ekvivalentan je IF u HLL programskom jeziku
- Jednakost se ispituje “=” za razliku od HLL gde se jednakost ispituje “==”
- U SQL- String podaci se stavljaju pod jednostrukim navodnicima za razliku od HLL jezika gde se koriste dvostruki navodnici

```
SELECT *  
FROM Klijent  
WHERE Uslov;
```

```
SELECT *  
FROM Klijent  
WHERE Objekt = '1234567890';
```


SQL RELACIONI OPERATORI i KOMBINOVANJE VIŠE USLOVA

```
SELECT *  
FROM Klijent  
WHERE Dobit > 100000;
```

```
SELECT *  
FROM Klijent  
WHERE Dobit > 100000  
AND Grad IN ('Lestovac', 'Niš');
```

>
<
>=
<=
<>

SQL – LIKE ISKAZ

- Komanda se koristi kada radimo operacije sa stringovima
- Za razliku od operacije “=” koja traži tačno mečovanje komanda Like nam dozvoljava veću fleksibilnost i efikasnost prilikom upoređivanja
- Wildcard pretraga
 - % mečuje više bilo kojih karaktera
 - _ mečuje jedan bilo koji karakter

Klijent
KlijentID (PK)
Ime
Prezime
Email
Telefon
Adresa1
Dobit

```
SELECT *  
FROM Klijent  
WHERE Email LIKE '%.rs';
```

SQL – NULL POLJA

Klijent
KlijentID (PK)
Ime
Prezime
Email
Telefon
Adresa1
Dobit
...

```
SELECT *  
FROM Klijent  
WHERE Email IS NULL;
```

```
SELECT *  
FROM Klijent  
WHERE Email IS NOT NULL;
```

SQL – SORTIRANJE REZULTATA

- Želimo da nam se prvo prikaže najveća cena a na kraju najmanja
- Default sortiranje je u rastućem redosledu

```
SELECT Opis, Cena, Boja  
FROM Proizvod
```

Opis	Cena	Boja
Produžni kabl	500	Crna
Punjač Baterija	4500	Crna
Slušalice	3000	Srebrne
Zvučnici	4000	Bela
...

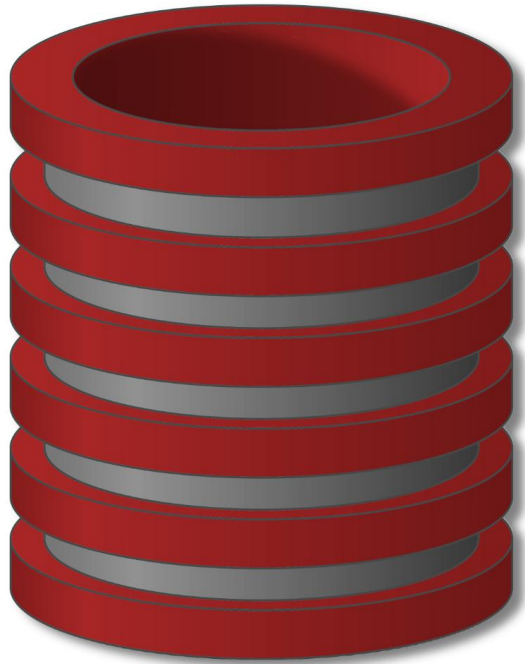
```
SELECT Opis, Cena, Boja  
FROM Proizvod  
ORDER BY Cena DESC
```

Opis	Cena	Boja
Punjač Baterija	4500	Crna
Zvučnici	4000	Bela
Slušalice	3000	Srebrne
Produžni kabl	500	Crna
...

SQL – SORTIRANJE NA OSNOVU VIŠE KRITERIJUMA

```
SELECT Opis, Cena, Boja  
FROM Proizvod  
ORDER BY Cena DESC, ,
```

Opis	Cena	Boja	Kategorija
Punjač Baterija	4500	Crna	Elektronika
Lego kockice	4000	Bela	Igračka
Slušalice	3000	Srebrne	Elektronika
SQL za 21 dan	3000	Crna	Učbenik
...	



SQL

Osnovni SELECT Iskaz

SQL: Basic SELECT

```
select A1, A2, ..., An
From   R1, R2, ..., Rm
where  uslov
```

Primer: Prijava studenata baza podataka

Fakultet (Fime, Grad, BrojMesta)
Student (Sid, Sime, Prosek, Vskole)
Prijava (Sid, Fime, Smer, Odluka)

Student

Sid	Sime	Prosek	Vskole

Prijava

Sid	Fime	Smer	Odluka

Fakultet

Fime	grad	BrojMesata

Zadatak:

Prikazati sid, ime i prosek studenata koji imaju prosek veći od 3.6

```
select Sid, Sime, Prosek  
From Student  
where Prosek > 3.6
```

Student

Sid	Sime	Prosek	Vskole

Prijava

Sid	Fime	Smer	Odluka

Fakultet

Fime	grad	BrojMesata

Zadatak:

Prikazati imena i smerove za koje su se studenti prijavili

```
Select Sime, Smer  
From Student, Prijava  
Where Student.Sid=Prijava.Sid
```

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

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

Student

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

Prijava

Fime	grad	BrojMesata
VTS	Nis	400
ELFAK	Nis	500
ETF	Bg	500
FON	BG	450

Fakultet

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
Vskole<500 and
Smer="SRT" and
Fime="VTS"
```

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

Student

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

Prijava

Fime	grad	BrojMesata
VTS	Nis	400
ELFAK	Nis	500
ETF	Bg	500
FON	BG	450

Fakultet

Zadatak:

Prikazati imena fakulteta koji primaju više od 500 studenata i imaju prijavljene studente na smer Multimedije

```
select Fakultet.Fime
From Fakultet,Prijava
where Fakultet.Fime=Prijava.Fime and
      BrojMesta>500 and
      Smer="Multimedije"
```

Student

Sid	Sime	Prosek	Vskole

Prijava

Sid	Fime	Smer	Odluka

Fakultet

Fime	grad	BrojMesata

Zadatak:

Prikazati sid, imena, prosek, fakultete na koje se student prijavio i broj slobodnih mesta na tom fakultetu. Sortirati dobijene podatke po proseku u opadajućem redosledu a zatim broju mesta u rastućem redosledu.

```
select Student.Sid,Sime,Prosek,Prijava.Fime,BrojMesta
From Student,Prijava,Fakultet
where Student.Sid=Prijava.Sid and
       Prijava.Fime = Fakultet.Fime
Order by Prosek Desc, BrojMesta Asc
```

Student

Sid	Sime	Prosek	Vskole

Prijava

Sid	Fime	Smer	Odluka

Fakultet

Fime	grad	BrojMesata

Zadatak:

Prikazati imena svih studenata koji su se prijavili na smer koji sadrži u svom nazivu reč "bio"

```
select Sime
From Student, Prijava
where Student.Sid=Prijava.Sid and
       Smer like "%bio%"
```

KARAKTER	OPIS
%	međuje više bilo kojih karaktera
_	međuje jedan bilo koji karakter

Student

Sid	Sime	Prosek	Vskole

Prijava

Sid	Fime	Smer	Odluka

Fakultet

Fime	grad	BrojMesata

Zadatak:

Prikazati sve podatke i attribute iz tabele Student

```
select *  
From Student
```

Zadatak:

Prikazati sve podatke i attribute iz tabela Student i Prijava

```
select *  
From Student, Prijava
```

Student

Sid	Sime	Prosek	Vskole

Prijava

Sid	Fime	Smer	Odluka

Fakultet

Fime	grad	BrojMesata

Zadatak:

Prikazati sve podatke iz tabele student uključujući i novu kolonu koja će predstavljati skaliran prosek a dobiće se kao rezultat vrednosti iz kolona Prosek i Veličina škole. Ideja je da oni studenti koji dolaze iz većih srednjih škola imaju i veći prosek. Koristiti formulu: $(Vskole/500)*Prosek$.

```
select Sid,Sime,Prosek,vskole,(vskole/500)*Prosek as 'Optimizovan Prosek'  
From Student
```

Student

Sid	Sime	Prosek	Vskole

Prijava

Sid	Fime	Smer	Odluka

Fakultet

Fime	grad	BrojMesata