

Fakultet tehničkih nauka, DRA, Novi Sad

Predmet:

Struktura i organizacija baza podataka

Dr Ivan Luković,

Mr Slavica Aleksić

Milan Čeliković

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].

Tabela radnik

- **Mbr** - maticni broj radnika
- **Ime** - ime radnika
- **Prz** - prezime radnika
- **Sef** - maticni broj direktno nadređenog rukovodioca - radnika
- **Plt** - mesecni 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

Tabela projekt

- **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

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

Ažuriranje baze podataka

- **INSERT**
- **DELETE**
- **UPDATE**

Ažuriranje baze podataka

- INSERT – dodavanje nove torke

```
INSERT INTO <naziv_tabele>  
(<lista_obeležja) VALUES  
(<lista_konstanti) | SELECT ...
```

Ažuriranje baze podataka

- INSERT – dodavanje nove torke

```
insert into Radnik (mbr, ime, prz, plt,  
sef, god) values (201, 'Ana', 'Savic',  
30000, null, '18-aug-71');
```

```
insert into Projekat (spr, nap, ruk)  
values ('70', 'P1', 201);
```

```
insert into RadProj (mbr, spr, brc)  
values (201, '10', 5);
```


Ažuriranje baze podataka

- DELETE – brisanje postojećih torki

**DELETE [FROM] <naziv_tabele>
[WHERE (<uslov_selekcije>)]**

Ažuriranje baze podataka

- DELETE – brisanje postojećih torki

delete radnik;

delete radnik where mbr=701;

Ažuriranje baze podataka

- Isključiti radnike sa onih projekta gde im je angažovanje na projektu manje od prosečnog angažovanja radnika na svim projektima

```
delete from radproj  
where brc < (select avg(brc) from  
radproj)
```

Ažuriranje baze podataka

- UPDATE – modifikacija postojećih torki

UPDATE <naziv_tabele>

SET <obeležje>= <aritm_izraz>

{,<obeležje>= <aritm_izraz>}

[WHERE (<uslov_selekcije>)]

Ažuriranje baze podataka

- UPDATE – modifikacija postojećih torki

```
update radnik  
set plt = plt*1.2;
```

```
update radnik  
set plt = plt*1.2  
where mbr = 201;
```

Ažuriranje baze podataka

- Svim radnicima koji rade na projektu sa šifrom 10 povećati premiju za 500

update radnik

set pre = isnull(pre,0) + 500

**where mbr in (select mbr from radproj
where spr = 10)**

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 Microsoft SQL servera.
- nazivi nisu case sensitive.

SQL tipovi podataka

Data	TypeDescription
– VARCHAR(size)	niz karaktera promenljive dužine, maksimalne dužine <i>size</i> ; minimalna dužina je 1, maksimalna je 8000
– CHAR(size)	Niz karaktera fiksne dužine od <i>size</i> bajtova; default i minimalna dužina je 1, maksimalna dužina je 8000
– NVARCHAR(size)	Unicode niz karaktera promenljive dužine, maksimalne dužine <i>size</i> ; minimalna dužina je 1, maksimalna je 4000
– NCHAR(size)	Unicode niz karaktera fiksne dužine od <i>size</i> bajtova; default i minimalna dužina je 1, maksimalna dužina je 4000
– NUMERIC(p,s)	broj ukupnog broja cifara p, od čega je s cifara iza decimalnog zareza; p može imati vrednosti od 1 do 38
– BINARY	binarni sadržaj do 8000 bajtova
– INT	
– REAL, FLOAT	
– DATETIME	vrednosti za vreme i datum
– TEXT, NTEXT	veliki text do $2^{31}-1$ karaktera
– IMAGE	veliki binarni sadržaj do $2^{31}-1$ bajtova

Kreiranje tabele

```
CREATE TABLE [šema.]<naziv_tabele>  
(<naziv_kolone> <tip_podatka> [DEFAULT izraz] [, ...]  
[CONSTRAINT constraint_name PRIMARY KEY  
(col1[,...])]  
[CONSTRAINT constraint_name FOREIGN KEY  
(col1[,...]) references table_name (col1[,...])]);
```

- 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

Kreiranje tabele

organizacija({Orgid, Orgnaziv, Mestoorg},{Orgid})

- **Orgid** – identifikacioni broj organizacije
- **Orgnaziv** – naziv organizacije
- **Mestoorg** – mesto u kojem se nalazi organizacija

Kreiranje tabele

```
create table organizacija  
(  
  orgid int not null,  
  orgnaziv varchar(20) null,  
  mestoorg varchar(20) null,  
  CONSTRAINT organizacija_pk  
  PRIMARY KEY (orgid)  
);
```

Kreiranje tabele

imenik ($\{\text{Mbr}, \text{Telbroj}\}, \{\text{Mbr}\}$)

imenik[Mbr] \subseteq radnik[Mbr]

- **Mbr** – matični broj radnika
- **Telbroj** – telefonski broj radnika

Kreiranje tabele

```
create table imenik  
(  
  mbr int not null,  
  telbroj varchar(10) not null,  
  CONSTRAINT imenik_pk PRIMARY KEY (mbr),  
  CONSTRAINT radnik_ti_fk FOREIGN KEY (mbr)  
  references radnik (mbr)  
);
```

Izmena definicije tabele

- **ALTER TABLE**

Alter table iskaz služi za:

- dodavanje nove kolone,
- modifikaciju postojeće kolone,
- definisanje podrazumevane vrednosti za novu kolonu,
- brisanje kolone.

ALTER TABLE

```
ALTER TABLE <naziv_tabele>  
ADD <naziv_kolone> <tip_podatka> [DEFAULT izraz]  
[, <naziv_kolone> <tip_podatka>]...;
```

```
ALTER TABLE <naziv_tabele>  
ALTER COLUMN <naziv_kolone> <tip_podatka>  
[DEFAULT izraz] [, <naziv_kolone> <tip_podatka>]...;
```

```
ALTER TABLE <naziv_tabele>  
DROP COLUMN <naziv_kolone>;
```

ALTER TABLE

- U tabelu imenik dodati obeležja mesto i adresa.

```
alter table imenik
```

```
add adresa varchar(20) null,  
mesto varchar(30) null
```

- U tabelu radnik dodati obeležje identifikacioni broj organizacione jedinice, a potom implementirati sledeće ograničenje referencijalnog integriteta

```
radnik[orgid] ⊆ organizacija[orgid]
```

```
alter table radnik
```

```
add orgid int null,  
constraint organizacija_fk Foreign Key (orgid)  
references organizacija (orgid)
```


ALTER TABLE

- Iz tabele organizacija ukloniti obeležje mestoorg

```
alter table organizacija  
drop column mestoorg
```

Brisanje definicije tabele

```
DROP TABLE <naziv_tabele>;
```

```
DROP TABLE IMENIK
```

Kreiranje, izmena i brisanje definicije pogleda

**CREATE VIEW <naziv_pogleda> [(alias
[, alias]...)]**

AS podupit;

- Podupit koji se koristi za definisanje pogleda može biti kompleksan, ali ne sme da sadrži ORDER BY.
- ORDER BY se koristi tek kada se prikazuju podaci iz pogleda.

Pogledi obezbeđuju sledeće prednosti

- ograničavaju pristup bazi podataka
- pojednostavljaju upite
- obezbeđuju nezavisnost podataka
- obezbeđuju višestruke poglede nad istim podacima
- mogu se brisati bez uklanjanja podataka u osnovnim tabelama.

Modifikacija pogleda

- Pogledi se modifikuju pomoću ALTER VIEW naredbe (nova definicija zamenjuje staru).
- Dakle, pogled može biti izmenjen bez brisanja postojećeg pogleda.
- Na primer, mogu se dodati alijasi za kolone u pogledu.

Kreiranje složenog pogleda

- Ukoliko se u upitu pomoću kog se kreira pogled nalaze skupovne funkcije (min, max, avg, sum, count) ili izrazi, u pogledu se moraju definisati alternativna imena za te kolone.

DML operacije sa pogledima

- DML (Data Manipulation Language) operacije se mogu primenjivati na jednostavnim pogledima.
- Ako pogled sadrži **skupovne funkcije**, **group by** kaluzulu, **distinct** rezervisanu reč ili **rownum** rezervisanu reč, vrsta iz pogleda se ne može izbrisati.
- Isto važi i za modifikaciju podataka, s tim što dodatno važi i da se **kolone definisane izrazima** ne mogu modifikovati (npr, salary*12).
- U pogled se ne mogu dodavati podaci ako pogled sadrži **skupovne funkcije**, **group by** kaluzulu, **distinct** rezervisanu reč, **rownum** rezervisanu reč, **kolonu koja je definisana izrazom**, ili **not null** kolonu u baznoj tabeli koja nije selektovana u pogledu.
- Dodavanjem vrednosti u pogled, one se dodaju direktno u baznu tabelu.

Brisanje pogleda

DROP VIEW pogled;

Pogled

- Kreirati pogled koji će prikazati mbr, ime i prz radnika i nazive projekata kojima rukovode.

```
CREATE VIEW rukovodioci (Mbr, Ime, Prezime,  
Projekt) AS
```

```
select r.mbr,ime, prz, nap  
from radnik r, projekat p  
where r.mbr=p.ruk
```

Pogled

```
SELECT Ime, Prezime, Projekat  
FROM rukovodioci
```

Pogled

- Kreirati pogled koji će prikazati mbr, ime i prz svih radnika i nazive projekata kojima rukovode. Ukoliko radnik ne rukovodi ni jednim projektom ispisati: ne rukovodi projektom.

```
CREATE VIEW rukovodioci_svi (Mbr, Ime, Prezime,  
Projekat) AS  
select r.mbr,ime, prz, isnull(nap, 'ne rukovodi  
projektom')  
from radnik r left outer join projekat p  
on r.mbr=p.ruk
```

Pogled

- Napraviti pogled koji će za svakog radnika prikazati njihova imena i prezimena kao i prezimena njihovih šefova ako ih imaju. Ako nema šefa ispisati: nema sefa.

```
CREATE VIEW radnici_sefovi (Ime, Prezime, Sef) AS  
select r1.ime, r1.prz, isnull(r2.prz, 'Nema sefa')  
from radnik r1 left outer join radnik r2  
on r1.sef=r2.mbr
```