

# OSNOVE BAZA PODATAKA

---

Standardni upitni jezik - SQL

# Naredbe za ažuriranje podataka

---

## □ DDL (Data Definition Language)

naredbe:

- INSERT
- UPDATE
- DELETE

# INSERT naredba

---

- ❑ Služi za unos novih n-torki u relaciju
- ❑ Opšti oblik naredbe:

```
INSERT [INTO] <naziv relacije>  
      [ ( <obl1>, <obl2>, ... )]  
VALUES  
      ( <vred1>, <vred2>, ... );
```

- ❑ obl1, obl2, ... - obeležja nove n-torke kojoj se definišu vrednosti
- ❑ vred1, vred2, ... – vrednosti obeležja nove n-torke

# Zadatak

---

- ❑ Dodati u relaciju NASTAVNIK podatke o nastavnicu Andjelic Andji, sa šifrom 110, zvanjem „R Prof“, koja radi od 2. marta 1984, prima platu 12000, nema dodatak i nema šefa.

```
insert into nastavnik
    (s_nas, prezime_ime, zvanje,
     s_dir, datzap, plata, dodatak)
values
    (110, 'Andjelic Andja', 'R PROF',
     null, '1984-03-02', 12000, null)
```

# UPDATE naredba

---

- ❑ Služi za izmenu sadržaja relacije
- ❑ Opšti oblik naredbe:

```
UPDATE <naziv_relacije>  
FROM obl1 = izraz1 [, obl2 = izraz2] ...  
[ WHERE <uslov> ];
```

```
UPDATE <naziv_relacije>  
FROM ( obl1 [, obl2] ) = ( podupit )  
[ WHERE <uslov> ];
```

- ❑ obl1, obl2, ... - obeležja relacije kojima se menja sadržaj
- ❑ izraz1, izraz2, ... – izrazi koji određuju vrednosti koje se pridružuju obeležjima obl1, obl2, ...

# UPDATE naredba

---

- ❑ Menja sadržaj navedenih obeležja, za one n-torke koje zadovoljavaju uslov u WHERE klauzuli
  - U slučaju kada se WHERE uslov izostavi, menja se sadržaj svih n-torki relacije
- ❑ Klauzula SET određuje koja obeležja podležu izmeni, i koje će im vrednosti biti dodeljene

# Zadatak

---

- Svim redovnim profesorima upisati dodatak od 1200 dinara.

```
update nastavnik  
set dodatak = 1200  
where zvanje = 'R PROF';
```

- Svim docentima čije prezime počinje na „P“ povećati platu za 5% i ukinuti dodatak.

```
update nastavnik  
set plata = plata * 1.05, dodatak = null  
where zvanje = 'DOCENT' and prezime_ime like 'P%';
```

# Zadatak

---

- Svim direktorima povećati dodatak za 10%.

```
update nastavnik  
set dodatak = dodatak * 1.10  
where s_nas in (select s_dir from nastavnik)
```



# Zadatak

---

- ❑ Nastavniku Savić Milanu postaviti nastavnika Radović Nikolu kao nadređenog (kao direktora).

```
update nastavnik
set s_dir = (select s_nas
              from nastavnik
              where prezime_ime = 'Radovic Nikola')
where prezime_ime = 'Savic Milan';
```

# Zadatak

---

- ❑ Ukinuti dodatak nastavnicima koji ne predaju ni jedan predmet.

```
update nastavnik  
set dodatak = null  
where s_nas not in (select s_nas  
                    from predaje);
```

# Zadatak

---

- Svim nastavnicima koji predaju više od jednog predmeta udvostručiti iznos dodatka.

```
update nastavnik
set dodatak = dodatak * 2
where s_nas in (select s_nas
                from predaje
                group by s_nas
                having count(s_pred) > 1);
```

# DELETE naredba

---

- ❑ Služi za brisanje n-torki iz relacija
- ❑ Opšti oblik naredbe:

```
DELETE FROM <naziv_relacije>  
[ WHERE <uslov> ];
```

- ❑ U slučaju da se ne zada WHERE uslov, naredba će obrisati sve n-torke u relaciji

# Zadatak

---

- ❑ Obrisati nastavnika sa šifrom 110.

```
delete from nastavnik  
where s_nas = 110;
```

- ❑ Obrisati nastavnika sa najvećom platom.

```
delete from nastavnik  
where plata >= all(select plata from nastavnik);
```

- Ova komanda se neće uspešno izvršiti

- ❑ Zašto?

- Obratiti pažnju na grešku koju prijavljuje SQL Server

The DELETE statement conflicted with the REFERENCE constraint ...

# Transakciona obrada

---

- **Transakcija** je vremenski uređeni niz nedeljivih radnju nad bazom podataka koje u celini ne remete uslove integriteta.
  - Transakcija predstavlja logičku jedinicu rada nad bazom podataka.
  - Transformiše jedno konzistentno stanje baze podataka u drugo takođe konzistentno stanje baze podataka.
    - Dozvoljeno je da se baza podataka između ta 2 konzistentna stanja nađe u nekonzistentnom stanju.
    - Bitno je da transakcija bude tako оформljena da iz bilo kojih razloga (otkaz hardvera ili softvera) ne ostavi trajno bazu podataka u nekonzistentnom stanju.

# Transakciona obrada

---

- Sistemi koji podržavaju **transakcionu obradu** moraju garantovati da, ukoliko se transakcijom vršilo ažuriranje baze podataka i iz bilo kojih razloga transakcija se nije normalno završila, budu poništena sva ažuriranja delimično izvršenih transakcija.

# Transakciona obrada

---

- ❑ Transakciona obrada odvija se pod kontrolom **administratora transakcija** kao dela programske podrške.
  - Administratoru transakcija možemo zadavati sledeće komande:
    - ❑ BEGIN TRANSACTION
    - ❑ COMMIT TRANSACTION
    - ❑ ROLLBACK TRANSACTION
  - Ove naredbe se zadaju direktno od strane programera, ili automatski od strane SUBP-a.



# Transakciona obrada

---

- ❑ BEGIN TRANSACTION
  - Signalizira administratoru transakcija o početku bloka transakcione obrade podataka.
- ❑ COMMIT TRANSACTION
  - Signalizira administratoru transakcija o uspešnom završetku transakcije.
    - ❑ Baza podataka se ponovo nalazi u konzistentnom stanju i sva dejstva transakcije (privremena ažuriranja) mogu biti trajno preneti na bazu podataka
- ❑ ROLLBACK TRANSACTION
  - Signalizira administratoru transakcija o neuspešnom završetku transakcije.
    - ❑ U tom slučaju je potrebno sva dejstva takve transakcije nad bazom podataka poništiti.

# Zadatak

---

- Svim docentima povećati platu za 10%. Takođe im povećati i broj časova za 2, na predmetima koje oni predaju. Svu obradu sprovesti transakciono.

```
begin transaction;
```

```
update nastavnik  
set plata = plata * 1.10  
where zvanje = 'Docent';
```

```
update predaje  
set casova = casova + 2  
where s_nas in (select s_nas  
                from nastavnik where zvanje = 'Docent');  
commit transaction;
```

# Domaći

---

## □ Kreirati tabele:

STUDENT ( S\_STUD, IME, PRZ, BR\_INDEKSA)

POLAZE ( S\_STUD, S\_PRED, BR\_BODOVA, OCENA )

Obeležje	Značenje	Dozvoljene nedostajuće vrednosti
S_STUD	Šifra studenta (koristi se kao identifikator studenta)	NE
IME	Ime studenta	NE
PRZ	Prezime studenta	NE
BR_INDEKSA	Broj indeksa studenta	NE
S_PRED	Šifra predmeta koji je student polagao	NE
BR_BODOVA	Broj bodova koji je student osvojio na predmetu	DA
OCENA	Ocena studenta iz predmeta	DA

# Domaći

---

## □ Kreirati tabele:

STUDENT ( S\_STUD, IME, PRZ, BR\_INDEKSA)

POLAZE ( S\_STUD, S\_PRED, BR\_BODOVA, OCENA )

```
create table student (  
    s_stud          smallint primary key,  
    ime            varchar(15) not null,  
    prz            varchar(20) not null,  
    br_indeksa     varchar(10) not null  
);  
  
create table polaze (  
    s_stud          smallint foreign key references student (s_stud),  
    s_pred          smallint foreign key references predmet (s_pred),  
    br_bodova       smallint,  
    ocena           smallint,  
    primary key (s_stud, s_pred)  
);
```

# Domaći

---

- Popuniti tabele STUDENT i POLAZE podacima:

```
insert into student values (1000, 'Sava', 'Savic', 'SR1/2021');
insert into student values (2000, 'Mirko', 'Miric', 'SR2/2021');
insert into student values (3000, 'Marina', 'Maric', 'SR3/2021');
insert into student values (4000, 'Javor', 'Jelic', 'SR4/2021');
insert into student values (5000, 'Goran', 'Goric', 'SR5/2021');
insert into student values (6000, 'Jovan', 'Ilic', 'SR6/2021');
insert into student values (100, 'Marko', 'Markovic', 'SR7/2019');
```

```
insert into polaze values (1000, 2, 87, 9);
insert into polaze values (4000, 2, 40, 5);
insert into polaze values (6000, 2, 57, 6);
insert into polaze values (100, 2, 71, 8);
insert into polaze values (1000, 3, null, null);
insert into polaze values (4000, 3, 17, 5);
insert into polaze values (5000, 3, 77, 8);
insert into polaze values (1000, 1, 80, 8);
insert into polaze values (3000, 1, 97, 10);
```

# Domaći

---

- ❑ Studentu koji se zove Javor Jelić povećati broj bodova iz predmeta Osnove Računarstva na 51, da bi imao dovoljno za ocenu 6.
  - Ažurirati i ocenu u istoj naredbi.
- ❑ Obrisati sve predmete koje niko ne predaje i niko ne polaže.
  - Sprovesti ažuriranje u posebnoj transakciji kako bi naknadno mogli poništiti ove promene.