

Baze podataka 1

Vežbe – SQL

Sadržaj

- Rad u učionici
- SQL - uvod i primer
- Jezik za definiciju podataka (DDL)
- Jezik za manipulaciju podacima (DML)
- Upitni jezik – SELECT naredba

Rad u učionici

Rad u učionici

- Baze podataka
 - studentska korisnička šema (user schema)
 - pod nazivom indXY
 - username: indXY
 - password: ftn
 - Gde je ind oznaka studijskog programa, X broj indeksa, a Y godina upisa

Rad u učionici

- Podaci potrebni za konektovanje na bazu

	MI A2-1, MI A2-2, MI A2-3	Učionice računarskog centra	Kod kuće
Username	indXY ¹	indXY ¹	*2
Password	ftn	ftn	*2
Role	default	default	default
Host Name	192.168.18.9	192.168.7.204	localhost
Port	1522	1521	1521
Oracle SID	db2016	bp1	
Service name			xepdb1

1 – ind oznaka studijskog programa, X broj indeksa, a Y godina upisa

2 – username i password koji su postavljeni tokom izvršavanja skripta za kreiranje korisnika

SQL - uvod i primer

Uvod

- SQL (engl. *Structured Query Language*)
 - 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
- 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
- SQL se javlja u formama:
 - interaktivnog jezika sistema za upravljanje bazama podataka
 - ugrađenog jezika u jezik III generacije
 - sastavnog dela jezika IV generacije

Uvod

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

Primer – relaciji model

- $\text{Radnik}(\{\text{Mbr, Ime, Prz, Sef, Plt, God, Pre}\}, \{\text{Mbr}\})$,
 - $\text{Projekat}(\{\text{Spr, Nap, Nar, Ruk}\}, \{\text{Spr}\})$,
 - $\text{Radproj}(\{\text{Spr, Mbr, Brc}\}, \{\text{Spr} + \text{Mbr}\})$,
-
- $\text{Radproj}[\text{Mbr}] \subseteq \text{Radnik}[\text{Mbr}]$,
 - $\text{Radproj}[\text{Spr}] \subseteq \text{Projekat}[\text{Spr}]$,
-
- $\text{Projekat}[\text{Ruk}] \subseteq \text{Radnik}[\text{Mbr}]$,
 - $\text{Null}(\text{Projekat}, \text{Ruk}) = \perp$
-
- $\text{Radnik}[\text{Sef}] \subseteq \text{Radnik}[\text{Mbr}]$,
 - $\text{Null}(\text{Radnik}, \text{Sef}) = \text{T}$

Tabela radnik

- Mbr - matični 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

Tabela projekat

- Spr - šifra projekta
 - Nap - naziv projekta
 - Nar - naručilac projekta
 - Ruk - rukovodilac projekta
-
- Obeležja Spr i Ruk ne smeju imati null vrednost, dok obeležje Nap mora imati jedinstvenu vrednost

Tabela radproj

- Spr - šifra projekta
 - Mbr - matični broj radnika
 - Brc - broj časova nedeljnog angažovanja na projektu
-
- Sva tri obeležja ne smeju da imaju null vrednost

Jezik za definiciju podataka (DDL)

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

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 i
- nazivi nisu case sensitive.

SQL tipovi podataka

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

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)  
);
```

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)  
);
```

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)  
);
```

Tabela faze_projekta – Zadatak za vežbu

- Kreirati tabelu faze_projekta
 - faze_projekta({Spr , Sfp, Rukfp, Nafp, Datp}, {Spr+ Sfp})
 - faze_projekta[*Spr*] \subseteq projekat[*Spr*],
 - faze_projekta[*Rukfp*] \subseteq radnik[*Mbr*]
- Sfp - šifra faze projekta,
- Spr - sifra projekta,
- Rukfp - rukovodilac faze projekta,
- Nafp - naziv faze projekta,
- Datp - datum početka faze projekta
- Obeležja Spr i Sfp ne smeju imati null vrednost.
- Obeležje Nafp mora imati jedinstvenu vrednost.

Tabela faze_projekta – Zadatak za vežbu

```
CREATE TABLE faze_projekta (  
    Spr integer not null,  
    Sfp integer not null,  
    Rukfp integer,  
    Nafp varchar2(20),  
    Datp date,  
    CONSTRAINT faze_projekta_PK PRIMARY KEY (spr, sfp),  
    CONSTRAINT faze_projekta_fk1 FOREIGN KEY (spr) REFERENCES projekat(spr),  
    CONSTRAINT faze_projekta_fk2 FOREIGN KEY (rukfp) REFERENCES radnik(mbr),  
    CONSTRAINT faze_projekta_uk UNIQUE(nafp)  
);
```

Izmena definicije tabele

- ALTER TABLE
- Alter table iskaz služi za:
 - dodavanje nove kolone,
 - modifikaciju postojeće kolone,
 - definisanje podrazumevane vrednosti za novu kolonu,
 - brisanje kolone i
 - dodavanje oraničenja.

ALTER TABLE

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

```
ALTER TABLE <naziv_tabele>  
MODIFY (<naziv_kolone> <tip_podatka> [DEFAULT izraz]  
    [, <naziv_kolone> <tip_podatka>]...);
```

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

```
ALTER TABLE <naziv_tabele>  
ADD CONSTRAINT <naziv_ogranicenja>  
<definicija_ogranicenja>;
```


Izmena definicije tabele – Zadatak za vežbu

- U tabelu faze_projekta dodati atribut:
 - Datz - datum završetka faze projekta
 - Datz ne sme biti manji od Datp

```
ALTER TABLE faze_projekta
  ADD datz date
  ADD CONSTRAINT dat_ch CHECK (datp<=datz);
```

```
ALTER TABLE faze_projekta
  ADD(datz date, CONSTRAINT dat_ch CHECK (datp<=datz));
```

```
ALTER TABLE faze_projekta
  ADD datz date;
ALTER TABLE faze_projekta
  ADD CONSTRAINT dat_ch CHECK (datp<=datz);
```

Brisanje definicije tabele

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

Brisanje definicije tabele – Zadatak za vežbu

- Izbrisati tabelu faze_projekta.

```
DROP TABLE faze_projekta;
```

Jezik za manipulaciju nad podacima (DML)

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', 'Jovic', 30000, null, '18-AUG-1971');
```

```
INSERT INTO projekat (spr, nap, ruk)
VALUES (90, 'P1', 201);
```

```
INSERT INTO radproj (mbr, spr, brc)
VALUES (201, 90, 5);
```

Skriptovi

- Pokrenuti odgovarajuće skriptove za punjenje baze podataka
 - radnik.sql
 - radproj.sql
 - projekat.sql

Ažuriranje baze podataka

- DELETE – brisanje postojećih torki

```
DELETE FROM <naziv_tabele>  
[WHERE (<uslov_selekcije>)];
```

Ažuriranje baze podataka

- DELETE – brisanje postojećih torki

```
DELETE FROM radnik;
```

```
DELETE FROM radnik  
WHERE mbr = 701;
```

```
DELETE FROM radproj;
```

Ažuriranje baze podataka – Zadatak za vežbu

- Probati brisanje torke koja je referencirana od strane neke druge torke.

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;
```

Transakcija

- Najmanja jedinica obrade podataka, takva da
 - prevodi bazu podataka iz jedno u drugo (ne nužno različito) konzistentno stanje, s obzirom na implementirana ograničenja
 - sadrži operacije upita ili/i operacije ažuriranja podataka u bazi podataka
- Efekti izvođenja transakcije se, na kraju, u celosti
 - potvrđuju (COMMIT) i tada postaju vidljivi ostalim korisnicima u sistemu, ili
 - poništavaju (ROLLBACK) i ostavljaju obrađivani deo baze podataka u stanju kakvo je važno neposredno pre početka njenog izvođenja

Upitni jezik – SELECT naredba

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
- sadržaj naveden unutar simbola „[“ i „]“ označava da je taj deo sintakse opcioni tj neobavezan

Upiti nad jednom tabelom – Primer

- Izlistati sadržaj svih tabela

```
SELECT * FROM radnik;
```

```
SELECT * FROM projekat;
```

```
SELECT * FROM radproj;
```

Upiti nad jednom tabelom – Zadatak za vežbu

- Prikazati imena i prezimena svih radnika.

```
SELECT ime, prez  
FROM radnik;
```

DISTINCT – Zadatak za vežbu

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

- Izlistati različita imena radnika, tj. imena bez ponovljenih vrednosti.

```
SELECT DISTINCT ime  
FROM radnik;
```

DISTINCT – Zadatak za vežbu

- Izlistati različita imena i prezime radnika, tj. imena i prezimena radnika bez ponovljenih vrednosti.

```
SELECT DISTINCT ime, prz  
FROM radnik;
```