**SQL UPITNI JEZIK – TRANSAKCIJE**

Opšti oblik transakcije u SQL-u, sa parcijalnim snimanjem i otkazivanjem transakcije:

**BEGIN TRANSACTION**

SELECT ...

INSERT ...

SAVE TRANSACTION Ime1 **/\* parcijalno snimanje transakcije \*/**

UPDATE ...

SAVE TRANSACTION Ime2

DELETE ...

IF @@ERROR<>0

ROLLBACK TRANSACTION Ime2

ROLLBACK TRANSACTION Ime1

**ROLLBACK TRANSACTION**

ELSE

COMMIT TRANSACTION

ILI:

**BEGIN TRANSACTION /\* početak transakcije \*/**

SELECT ...

INSERT ... **/\* elementarne transakcije \*/**

UPDATE ...

DELETE ... **/\* SQL upiti i naredbe\*/**

IF @@ERROR=0

**COMMIT TRANSACTION /\* uspešan završetak \*/**

ELSE

ROLLBACK TRANSACTION **/\* neuspešan završetak \*/**

**Šema relacione baze podataka FAKULTET:**



**1. Napisati transakciju za evidentiranje studenta Višković Nenada, broj indeksa: SI 25/15, smera Softversko inzenjerstvo, rođenog 20.4.1995., stanuje u Despotovcu, u ulici Manastirska bb, ptt broj mesta je 13200.**

BEGINTRANSACTION

INSERTINTO **mesto**

VALUES(13200,'Despotovac')

INSERTINTO **student** /\*(prezime, ime, ocena,...)\*/

VALUES ('SI 25/15','Nenad','Viškoviæ','Manastirska bb',

'1995-4-20','SI',13200)

IF@@ERROR=0

COMMITTRANSACTION

ELSE

ROLLBACKTRANSACTION

**2. Napisati transakciju za evidentiranje odslušanog predmeta Baze podataka 1 za studenta Višković Nenada, sa brojem indeksa: SI 25/15. Prilikom pisanja transakcije uzeti u obzir da je predmet je već evidentiran u bazi podataka! ID predmeta odrediti na osnovu naziva.**

BEGINTRANSACTION

DECLARE **@idpred** int

SELECT **@idpred**=idpredmeta

FROM **predmet**

WHERE **nazivpredmeta**='Baze podataka 1'

INSERTINTO **slusao**

VALUES ('IT 65/15',@idpred)

IF@@ERROR=0

COMMITTRANSACTION

ELSE

ROLLBACKTRANSACTION

**3. Napisati transakciju za evidentiranje položenog ispita iz Baza podataka 1 za studenta Ekres Viktoriju, broj indeksa: IT 65/15, ocena je 10, bodova 100, datum polaganja je 5.5.2017., id premeta je 1, id nastavnika je 2. Redni broj polaganja ispita je 70.**

BEGINTRANSACTION

INSERTINTO **ispit**

VALUES ('70','IT 65/15','1','2','2017/5/5','10','100')

/\*VALUES ('IT 65/15','1','2','2017/5/5',10,100) autoincrement,identity\*/

IF@@ERROR=0

COMMITTRANSACTION

ELSE

ROLLBACKTRANSACTION

**4. Napisati transakciju za evidentiranje položenog ispita iz Projektovanja softvera za studenta Viktoriju Ekres, broj indeksa: IT 65/15. Ocena je 6, osvojeno 56 bodova, datum polaganja je 8.5.2017. Predmet pripada Katedri za Informacione tehnologije i predaje ga doc. dr Eleonora Brtka iz Zrenjanina, ptt je 23000, sa nedeljnim fond časova 2. Proveriti da li su nastavnik i predmet već upisani u bazu podataka i ukoliko nisu upisati ih. Redni broj polaganja, id predmeta i id nastavnika odrediti u okviru transakcije tako sto ce se najveći uneti broj povećati za 1.**

BEGINTRANSACTION

/\*upis mesta\*/

INSERTINTO **mesto**

VALUES (23000,'Zrenjanin')

/\*provera da li postoji nastavnik\*/

DECLARE **@idn** int

SELECT **@idn**=idnastavnika

FROM **nastavnik**

WHERE **ime**='Eleonora'and **prezime**='Brtka'

IF **@idn** ISNULL

BEGIN

/\*odredjivanje najveceg id nastavnika + 1\*/

DECLARE **@maksrb** INT

SELECT **@maksrb**=MAX(idnastavnika)FROM **nastavnik**

IF **@maksrb** ISNULL

SET **@maksrb**=1

ELSE

SET **@maksrb**=@maksrb+1

/\*upis nastavnika\*/

SET **@idn**=@maksrb

INSERTINTO **nastavnik**

VALUES(@idn,'Eleonora','Brtka','docent',null,null,null,23000)

END

/\*provera da li postoji predmet\*/

DECLARE **@idpred** int

SELECT **@idpred**=idpredmeta

FROM **predmet**

WHERE **nazivpredmeta**='Projektovanje softvera'

IF **@idpred** ISNULL

BEGIN

/\*odredjivanje najveceg id predmeta + 1\*/

DECLARE **@maksid** INT

SELECT **@maksid**=MAX(idpredmeta)FROM **predmet**

IF **@maksid** ISNULL

SET **@maksid**=1

ELSE

SET **@maksid**+=1

/\*odredjivanje id katedre za IT\*/

DECLARE **@idkat** int

SELECT **@idkat**=idkatedre

FROM **katedra**

WHERE **naziv**='Katedra za Informacione tehnologije'

/\*upis predmeta ukoliko ne postoji u bazi\*/

SET **@idpred**=@maksid

INSERTINTO **predmet**

VALUES(@idpred,'Projektovanje softvera',@idkat)

END

/\*odredjivanje rb polaganja\*/

DECLARE **@rbp** INT

SELECT **@rbp**=MAX(rbpolaganja)FROM **ispit**

IF **@rbp** ISNULL

SET **@rbp**=1

ELSE

SET **@rbp**=@rbp+1

/\*evidentiranje predavanja\*/

INSERTINTO **predaje**

VALUES (@idn,@idpred,2)

/\*evidentiranje odlusanog predmeta\*/

INSERTINTO **slusao**

VALUES ('IT 65/15',@idpred)

/\*upis ispita u bazu\*/

INSERTINTO **ispit**

VALUES(@rbp,'IT 65/15',@idpred,@idn,'2017/5/8',6,56)

IF@@ERROR=0

COMMITTRANSACTION

ELSE

ROLLBACKTRANSACTION

**5. Napisati transakciju za promenu adrese boravka studenta Nenada Viškovića, SI 25/15. Nova adresa boravka je: 22300 Stara Pazova, Beogradska 32/45.**

BEGIN TRANSACTION

INSERT INTO mesto (naziv, ptt)

VALUES ('Stara Pazova', 22300);

UPDATE student

SET prezime='Višković', ulicaBrStanuje='Beogradska 32/45', pttStanuje=22300

WHERE brojIndeksa='SI 25/15' /\*and ime='Nenad' and prezime='Višković';\*/

IF @@ERROR>0

ROLLBACK TRANSACTION

ELSE

COMMIT TRANSACTION

**6. Napisati transakciju za brisanje svih podataka (ispita, odslušanih predmeta) o studentu Nenadu Viškoviću, broj indeksa SI 25/15 iz baze podataka. Relacije između su sa restriktivnim pravilom za očuvanje referencijalnog integriteta.**

BEGIN TRANSACTION

DELETE student

WHERE brojIndeksa='SI 25/15'/\*ukoliko su pravila za

ocuvanje referencijalnog integriteta postavljena kaskadno\*/

IF @@ERROR>0

ROLLBACK TRANSACTION

ELSE

COMMIT TRANSACTION

BEGIN TRANSACTION

/\*ukoliko su pravila za

ocuvanje referencijalnog integriteta postavljena restrivno\*/

DELETE ispit

WHERE brojIndeksa='SI 25/15'

DELETE slusao

WHERE brojIndeksa='SI 25/15'

DELETE student

WHERE brojIndeksa='SI 25/15'

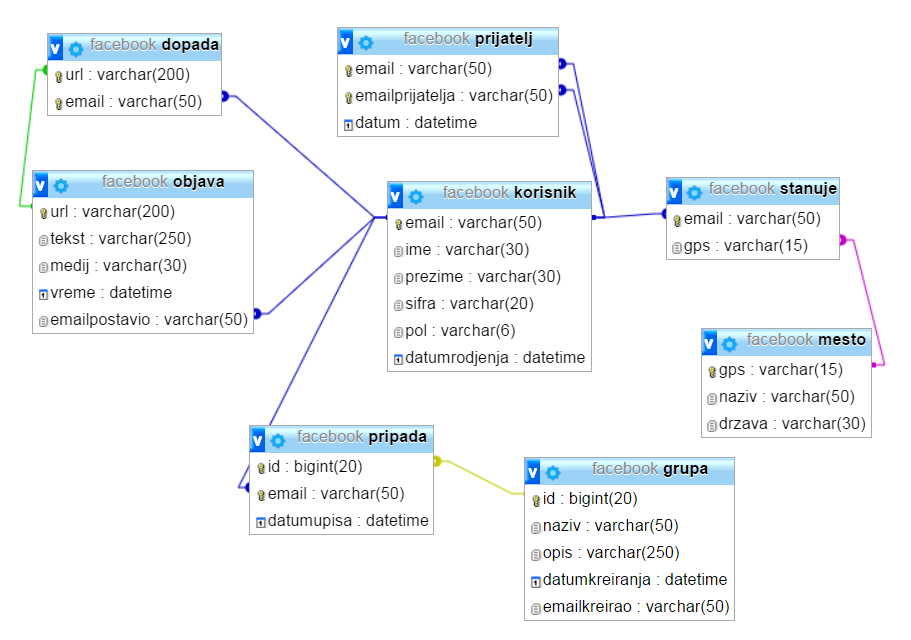
IF @@ERROR>0

ROLLBACK TRANSACTION

ELSE

COMMIT TRANSACTION

**Primer baze podataka Facebook (Relaciona šema iz MySQL softvera za rukovanje bazama podataka):**



**ZADACI ZA AKTIVNOST NA ČASOVIMA br. 4**

1. Napisati transakciju za registrovanje novog korisnika (svaki student za sebe, sa adresom boravka).

2. Napisati transakciju za formiranje grupe pod nazivom „SQL upitni jezik“ i evidentirati se (svako za sebe) u okviru te grupe. ID grupe odrediti u transakciji kao najveći uneti ID uvećan za 1. Ukoliko je to prva grupa ID treba da ima vrednost 1. Datum kreiranja grupe treba da je današnji (tekući) datum.