

Laboratorij programskog inženjerstva i informacijskih sustava 2

odjeljak Sustavi baza podataka

Laboratorijske vježbe 3. dio Transakcije

Cilj ovih laboratorijskih vježbi je upoznavanje s osnovama upravljanja granicama transakcija u pohranjenoj proceduri i klijentskoj aplikaciji (JAVA+JDBC API).

1. ZADATAK

- osmisлити vrlo jednostavan model baze podataka i jednostavnu transakciju. Kao mjera "jednostavnosti" može poslužiti transakcija koja je realizirana pohranjenom procedurom u domaćoj zadaći koja se bavila temom transakcija
- napisati INSERT naredbe kojima će se testne relacije napuniti s onoliko n-torki koliko je potrebno da bi se obavila testiranja opisana u nastavku vježbi (pretpostavlja se da će biti dovoljno tek nekoliko n-torki)
- uočiti da primjer koji se koristio na domaćoj zadaći ne prikazuje "neku realnu transakciju koja se koristi u bankarstvu". Radi se o drastično pojednostavljenom primjeru koji je ipak dovoljno dobar (i logičan) za prikaz najvažnijih koncepata. Jednako tako, od vas se očekuje da napravite vrlo jednostavan, ali logičan primjer transakcije. Npr. nelogično bi bilo definirati transakciju koja se sastoji od operacija unosa n-torke u relaciju osoba i brisanja n-torke iz relacije zrakoplov (zašto bi se te dvije operacije trebale obavljati u okviru jedne transakcije?)
- model baze i transakcije mora biti potpuno različit od primjera na domaćoj zadaći
- transakcija treba:
 - sadržavati najmanje dvije SQL naredbe koje u određenim uvjetima mogu izazvati pogrešku zbog narušavanja integritetskih ograničenja
 - najmanje jedna od pogrešaka koje mogu biti izazvane narušavanjem u bazu podataka ugrađenih integritetskih ograničenja mora pozivajućem programu biti dojavljena na hrvatskom jeziku
- prije nego se pristupi modeliranju baze i transakcije, pogledati zadatke u nastavku

Što treba predati kao rezultat 1. zadatka:

- a) sliku i vrlo kratki opis ER modela
- b) vrlo kratki tekst kojim se opisuje transakcija
- c) skripta sa SQL naredbama za kreiranje relacija u sustavu IBM IDS, uključujući integritetska ograničenja koja su bitna u nastavku vježbi, te INSERT naredbe za unos testnih n-torki

2. ZADATAK

- a) napisati IBM IDS (Informix) SPL pohranjenu proceduru koja realizira transakciju iz 1. zadatka pri čemu procedura mora samostalno upravljati granicama transakcije. Također, procedura mora "presretati" i prilagoditi tekst barem jedne od pogrešaka koje se mogu pojaviti tijekom izvršavanja transakcije. Prilagodba teksta pogreške znači: zamijeniti originalni tekst pogreške nekim tekstom s korisniku razumljivom porukom, poput teksta "Nedopušteni minus" koji se koristio u primjeru u domaćoj zadaći
- b) napisati naredbe za testiranje transakcije pozivima procedure iz interaktivnog alata (npr. Server Studio)
 - jedan poziv procedure koji će rezultirati uspješnim obavljanjem transakcije
 - jedan poziv procedure koji će rezultirati prekidom transakcije pri čemu će SQL klijent (npr. Server Studio) dojaviti broj pogreške i tekst pogreške koju je metoda "presrela" i prilagodila (npr. tekst na hrvatskom umjesto na engleskom)
 - jedan poziv procedure koji će rezultirati prekidom transakcije pri čemu će SQL klijent dojaviti broj i tekst pogreške koja je iz metode proslijeđena bez izmjene (originalnu pogrešku)
- c) nakon provođenja navedenih testova, modificirati proceduru tako da SUBP narušavanje bilo kojeg integritetskog ograničenja ugrađenog u bazu podataka provjerava tek na kraju transakcije, neposredno prije potvrđivanja transakcije. Po potrebi prilagoditi programski kôd procedure u dijelovima koji upravljaju pogreškama. Ispravnost modificirane procedure provjeriti ponavljanjem testova iz prethodne točke

Što treba predati kao rezultat 2. zadatka:

- a) SQL skripta s naredbama za kreiranje procedure i naredbama za testiranje procedure
- b) SQL skripta s naredbom za kreiranje modificirane procedure u kojoj se ugrađena integritetska ograničenja provjeravaju tek na kraju transakcije

3. ZADATAK

- a) napisati java program koji sadrži metodu koja koristi JDBC API i koja realizira transakciju na isti način kako je ta transakcija realizirana IDS SPL procedurom u 2. zadatku. Pri tome:
 - o barem na jednom mjestu u metodi primijeniti PreparedStatement objekt (u primjerima u domaćoj zadaći se koristio isključivo Statement objekt)
 - o uočiti da se pohranjena procedura iz prethodnog zadatka ne koristi
- b) pripremiti testove kojima će se prikazati
 - o jedan poziv metode koji će rezultirati uspješnim obavljanjem transakcije
 - o jedan poziv metode koji će rezultirati prekidom transakcije pri čemu će java klijent dojaviti broj pogreške i tekst pogreške koju je metoda "presrela" i prilagodila (npr. tekst na hrvatskom umjesto na engleskom)
 - o jedan poziv metode koji će rezultirati prekidom transakcije pri čemu će java klijent dojaviti broj i tekst pogreške koja je iz metode proslijeđena bez izmjene (originalnu pogrešku)
- c) nakon provođenja navedenih testova, modificirati programski kôd iz a) dijela zadatka tako da SUBP narušavanje bilo kojeg integritetskog ograničenja ugrađenog u bazu podataka provjerava tek na kraju transakcije, neposredno prije potvrđivanja transakcije. Ispravnost modificirane metode provjeriti ponavljanjem testova iz prethodne točke
- d) modificirati programski kôd iz a) dijela zadatka tako da transakcija nakon prve obavljene naredbe koja mijenja podatke u bazi podataka (izmjena, unos ili brisanje podataka) bude prekinuta zbog "neočekivanog prekida" klijentskog programa (a ne eksplicitnim pozivom metode rollback). U svrhu simulacije "neočekivanog prekida" programa može se koristiti java metoda koja je po nazivu jednaka funkciji iz programskog jezika C koja se koristi za trenutno prekidanje izvršavanja programa. Pripremiti i provesti test kojim će se dokazati očekivano ponašanje sustava u slučaju "neočekivanog prekida" klijentskog programa.

Što treba predati kao rezultat 3. zadatka:

- a) java program
- b) uputu kako obaviti navedena tri testa (npr. tri naredbe za pokretanje java programa iz naredbene linije)
- c) modificirani java program u kojem se ugrađena integritetska ograničenja provjeravaju tek na kraju transakcije
- d) modificirani java program u kojem se simulira "neočekivani prekid" klijentskog programa

4. ZADATAK

- model baze podataka iz prvog zadatka implementirati u sustavu PostgreSQL

Što treba predati kao rezultat 4. zadatka:

- skripta sa SQL naredbama za kreiranje relacija u sustavu PostgreSQL, uključujući integritetska ograničenja, te INSERT naredbe za unos testnih n-torki

5. ZADATAK

- a) napisati PL/pgSQL funkciju koja realizira transakciju iz 1. zadatka (uočite, PostgreSQL ne dopušta pokretanje transakcije unutar funkcije već se funkcija sama po sebi ponaša kao "subtransakcija". Uspješnim završetkom funkcije "subtransakcija" se potvrđuje, neuspješnim poništava). Također, funkcija mora "presretati" i prilagoditi tekst barem jedne od pogrešaka koje se mogu pojaviti tijekom izvršavanja. Prilagodba teksta pogreške znači: zamijeniti originalni tekst pogreške nekim tekstom s korisniku razumljivom porukom, poput teksta "Nedopušteni minus" koji se koristio u primjeru u domaćoj zadaći
- b) napisati naredbe za testiranje transakcije pozivima procedure iz interaktivnog alata (npr. Server Studio)
 - jedan poziv funkcije koji će rezultirati uspješnim obavljanjem transakcije
 - jedan poziv funkcije koji će rezultirati prekidom transakcije pri čemu će SQL klijent (npr. Server Studio) dojaviti broj pogreške i tekst pogreške koju je metoda "presrela" i prilagodila (npr. tekst na hrvatskom umjesto na engleskom)
 - jedan poziv funkcije koji će rezultirati prekidom transakcije pri čemu će SQL klijent dojaviti broj i tekst pogreške koja je iz metode proslijeđena bez izmjene (originalnu pogrešku)

Što treba predati kao rezultat 5. zadatka:

- a) SQL skripta s naredbama za kreiranje procedure i naredbama za testiranje procedure

6. ZADATAK

- a) napisati java program koji sadrži metodu koja koristi JDBC API i koja u sustavu PostgreSQL realizira transakciju na isti način kako je ta transakcija realizirana IDS SPL procedurom u 2.a) zadatku. Pri tome:
 - barem na jednom mjestu u metodi primijeniti PreparedStatement objekt (u primjerima u domaćoj zadaći se koristio isključivo Statement objekt)
 - uočiti da se pohranjena procedura iz prethodnog zadatka ne koristi
- b) pripremiti testove kojima će se prikazati
 - jedan poziv metode koji će rezultirati uspješnim obavljanjem transakcije
 - jedan poziv metode koji će rezultirati prekidom transakcije pri čemu će java klijent dojaviti broj pogreške i tekst pogreške koju je metoda "presrela" i prilagodila (npr. tekst na hrvatskom umjesto na engleskom)
 - jedan poziv metode koji će rezultirati prekidom transakcije pri čemu će java klijent dojaviti broj i tekst pogreške koja je iz metode proslijeđena bez izmjene (originalnu pogrešku)
- c) modificirati programski kôd iz a) dijela zadatka tako da transakcija nakon prve obavljene naredbe koja mijenja podatke u bazi podataka (izmjena, unos ili brisanje podataka) bude prekinuta zbog "neočekivanog prekida" klijentskog programa (a ne eksplicitnim pozivom metode rollback). U svrhu simulacije "neočekivanog prekida" programa može se koristiti java metoda koja je po nazivu jednaka funkciji iz programskog jezika C koja se koristi za trenutno prekidanje izvršavanja programa. Pripremiti i provesti test kojim će se dokazati očekivano ponašanje sustava u slučaju "neočekivanog prekida" klijentskog programa.

Što treba predati kao rezultat 6. zadatka:

- a) java program
- b) uputu kako obaviti navedena tri testa (npr. tri naredbe za pokretanje java programa iz naredbene linije)
- c) modificirani java program u kojem se simulira "neočekivani prekid" klijentskog programa

Dodatna literatura:

- PostgreSQL 9.4.0 Documentation, Chapter 40. PL/pgSQL - SQL Procedural Language
- IBM Informix JDBC Driver, Programmer's Guide
<http://publibfp.dhe.ibm.com/epubs/pdf/c2735543.pdf>
- The PostgreSQL JDBC Interface
<https://jdbc.postgresql.org/documentation/94/index.html>
- kao brzi priručnik za JDBC mogu se koristiti materijali objavljeni uz domaću zadaću koja se bavila temom transakcija (*jdbcUvod.pdf*).

ZAVRŠNE NAPOMENE:

- programe, skripta, itd. priložiti kao zasebne datoteke
- sve ostalo (što je navedeno da treba predati kao rezultate 1. zadatka) predati u jednoj MS Word datoteci (ne .txt, ne .pdf,...).
 - ime datoteke treba biti: Lp3**Vašeprezime**.doc ili Lp3**Vašeprezime**.docx
 - napisati uredno zaglavlje (naslov, podaci o studentu, datum predaje, ...)
- uzeti u obzir da je urednost i organiziranost predanih rezultata jedan od elemenata koji će utjecati na ocjenu