

OBLIKOVANJE BAZA PODATAKA

Vježbe 11

Transakcije - uvod

- **Transakcija** (engl. *transaction*) predstavlja količinu posla koju baza treba odraditi kao cjelinu
- Postoje dva tipa transakcija:
 - Transakcije na nivou jedne naredbe
 - Svaka SQL naredba je transakcija sama po sebi
 - Ako zadamo DELETE za 5 redaka, ne može se desiti da se 2 retka obrišu, a 3 ne
 - Eksplicitne transakcije koje obuhvaćaju proizvoljan broj naredbi
 - Korisnik zadaje početak i kraj transakcije

Korisnost transakcija

- Transakcije rješavaju problem da je baza uvijek u **ispravnom (konzistentnom)** stanju
 - Svaka naredba se mora izvršiti do kraja ili uopće ne
 - Radimo UPDATE na 20 redaka
 - Ponekad treba osigurati da se niz SQL naredbi odradi kompletan ili da se ne odradi niti jedna naredba iz niza
 - Želimo nekome prebaciti 1.000 kuna na račun:
 - UPDATE koji skida 1.000 kuna s našeg računa
 - UPDATE koji stavlja 1.000 kuna na odredišni račun

Transakcijska datoteka

- Transakcijska datoteka omogućava postojanje transakcija
- Svaka relacijska baza se obavezno sastoji od **barem jedne transakcijske datoteke**
 - Sadržava povijest akcija rađenih nad bazom
 - Svaka akcija nad bazom se zapisuje o transakcijsku datoteku zajedno s podacima na koje je utjecala
 - Kod SQL Servera se ta datoteka naziva i **transakcijski zapisnik** (engl. *transactional log*) i ima ekstenziju **.ldf**
- Zbog performansi je dobro transakcijsku datoteku držati na drugom fizičkom disku
 - Na taj način se mogu paralelno zapisivati podaci i transakcijske aktivnosti

Važni trenuci u životu jedne transakcije

- **Početak** transakcije
- **Kraj** transakcije
 - **Potvrda** transakcije
 - Sve operacije koje čine transakciju su **uspješno** obavljene
 - **Odustajanje** od transakcije
 - Desila se jedna ili više grešaka i transakcija je **neuspješna**
 - Stanje baze vraća se na stanje prije početka transakcije
- Opcionalno postavljanje **kontrolne točke** (engl. *savepoint*)

T-SQL sintaksa za upravljanje transakcijama

- Početak transakcije:
BEGIN TRAN[SACTION]
- Potvrda transakcije:
COMMIT TRAN[SACTION] (ili samo **COMMIT**)
- Odustajanje od transakcije:
ROLLBACK TRAN[SACTION] (ili samo **ROLLBACK**)

T-SQL sintaksa za upravljanje transakcijama

- Dodatno, za rad s kontrolnim točkama koristimo:

- Postavljanje kontrolne točke:

SAVE TRAN[SACTION] ***naziv_kontr_točke***

- Vraćanje na kontrolnu točku:

ROLLBACK TRAN[SACTION] ***naziv_kontr_točke***

Primjer transakcije

- Svaka transakcija se odnosi na konekciju na kojoj je zadana
 - Otvorimo SSMS, napravimo novu konekciju na neku bazu i zadamo naredbe:



- Nakon kraja transakcije u bazi će ostati promjene napravljene naredbama 1, 2 i 4
- Učinak naredbe 3 je poništen vraćanjem na kontrolnu točku naziva spa

Sistemska funkcija @@TRANCOUNT

- Sistemska funkciju **@@TRANCOUNT** koristimo za provjeru jesmo li unutar transakcije
- Način rada:
 - **BEGIN TRAN** povećava **@@TRANCOUNT** za 1
 - **COMMIT TRAN** smanjuje **@@TRANCOUNT** za 1
 - **ROLLBACK TRAN** smanjuje **@@TRANCOUNT** na 0
- Povratak na kontrolnu točku ne mijenja **@@TRANCOUNT**

Transakcije (1)

Napravite tablicu Osoba s IDOsoba (PK i IDENTITY), Ime i Prezime.

1. U transakciji umetnite 3 zapisa u Osoba i na kraju odustanite od transakcije.
2. U transakciji umetnite 3 zapisa u Osoba i na kraju potvrdite transakciju.
3. U transakciji umetnite 1 zapis u Osoba i postavite kontrolnu točku. Umetnite još 1 zapis. Na kraju odustanite od transakcije.
4. U transakciji umetnite 1 zapis u Osoba i postavite kontrolnu točku. Umetnite još 1 zapis. Na kraju potvrdite transakciju.

Transakcije (2)

5. U transakciji umetnite 1 zapis u Osoba i postavite kontrolnu točku. Umetnite još 1 zapis i postavite kontrolnu točku. Na kraju odustanite od transakcije.
6. U transakciji umetnite 1 zapis u Osoba i postavite kontrolnu točku. Umetnite još 1 zapis i postavite kontrolnu točku. Na kraju potvrdite transakciju.
7. U transakciji umetnite 1 zapis u Osoba i postavite kontrolnu točku. Umetnite još 1 zapis i vratite se na kontrolnu točku. Na kraju odustanite od transakcije.
8. U transakciji umetnite 1 zapis u Osoba i postavite kontrolnu točku. Umetnite još 1 zapis i vratite se na kontrolnu točku. Na kraju potvrdite transakciju.

Transakcije (3)

9. Napišite proceduru za brisanje države. Neka procedura prima 1 parametar, IDDrzava. Transakciju vodite izvan procedure. Ispišite uspjeh ili neuspjeh.
- Pozovite 3 puta proceduru s vrijednostima parametara jednakim 50, 51 i 52.
 - Pozovite 3 puta proceduru s vrijednostima parametara jednakim 50, 51 i 1.
10. Napišite proceduru s 3 parametra za brisanje triju država. Transakciju vodite unutar procedure. Ispišite uspjeh ili neuspjeh.
- Pozovite proceduru s vrijednostima 50, 51 i 52.
 - Pozovite proceduru s vrijednostima 50, 51 i 1.

Transakcije (4)

11. Unutar vanjske transakcije pozovite prethodnu proceduru s vrijednostima 50, 51 i 52. Nakon toga odustanite od vanjske transakcije. Ispišite uspjeh ili neuspjeh. Je li brisanje napravljeno?
12. Osmislite XML za unos računa s tri stavke. Napišite proceduru za unos računa i unutar transakcije odradite umetanje računa i stavki. Pozovite proceduru.