

Baze podataka 2

I kolokvijum

15.12.2012.

Opis problema - putarina

(23 – boda, vreme rada 150 min)

Na autoputu Beograd-Niš se nalazi nekoliko kontrolnih rampi. Poznata su nam njihova imena i udaljenosti (kilometri) između svake dve od njih u oba smera. Potrebno je modelovati funkcionisanje dela sistema naplate korišćenja autoputa na nivou baze podataka. Svaka od kontrolnih rampi šalje podatke o ulasku ili izlasku određenog vozila sa autoputa. Te podatke beležimo u odgovarajućim tabelama i pomoću njih formiramo iznos za naplatu prilikom izlaska vozila sa autoputa. Iznos se dobija množenjem pređene udaljenosti u kilometrima sa cenom takse po jednom kilometru.

Tabele baze:

- **rampe(id,naziv)** - id kontrolne rampe i naziv (mesto);
- **vozila(reg_broj)** – registar svih vozila, smatrati da se samo ova vozila mogu naći na autoputu;
- **udaljenosti(odakle_id, dokle_id, udaljenost)** - id-jevi mesta i njihove udaljenosti; za svaki par mesta po dve, jer ne moraju nužno biti iste;
- **cene(cena,datum_vazenja)** - cene se mogu menjati tokom vremena; uzima se cena sa tekućim datumom važenja; ukoliko je vozilo stupilo na autoput (vreme ulaska) pre tekućeg (najnovijeg) datuma važenja, naplaćuje se po staroj ceni koja je bila u trenutku ulaska vozila;
- **promet(reg_broj, ulazak_id, datum_ulaska, izlazak_id, datum_izlaska, cena, iznos)** – za vozilo (prepoznavamo ga sa registarskim tablicama) promet se prati sa podacima koji govore gde je vozilo ušlo na autoput, gde je izašlo, vremena ulaska i izlaska, cena koja je važila u tom trenutku, i konačni iznos koji je potrebno platiti kada se registruje izlazak. Prilikom ulaska, definišu se podaci o registarskom broju vozila, mestu ulaska, datumu ulaska i važećoj ceni takse.
- **naplaceno(reg_broj, ulazak_id, datum_ulaska, izlazak_id, datum_izlaska, cena, iznos)** - podaci iz prometa, identični matičnim, ali arhivski prebačeni u ovu tabelu jer nisu aktuelni.

T-SQL skripte (10 bodova)

Potrebno je napisati skripte:

- **ulazak.sql** - slučajnim izborom bira mesto ulaska, registarski broj i generiše odgovarajući slog u prometu na svakih 10 sekundi. Jedna skripta staje nakon generisanih 10000 slogova.
- **naplata.sql** - od dostupnih vozila u prometu (koja još nisu izašla) slučajnim izborom odabere jedno vozilo. Za to vozilo se generiše izlazak i registruje plaćanje u datom slogu - update datog sloga odgovarajućim podacima. Vremena izlazaka beležiti kao da su se desili 120 ± 60 minuta posle ulaska. Skripta izvršava generisanje izlaska na svakih 10 sekundi i staje nakon obrađenih svih slogova.
- **analiza.sql** – na svakih 10 minuta proverava broj vozila koja su trenutno u prometu (ušli na autoput, a još nisu izašli).
- **arhiva.sql** – na svakih 5 minuta sve slogove iz tabele promet koji su naplaćeni prebacuje u arhivsku tabelu naplaceno. Prebacivanje vršiti u chunk-ovima od 5 slogova.

Transakcije (13 bodova - 4 + 9)

Uposliti transakcije (uz isolation level i savepoint mehanizme) u svim T-SQL skriptama kako bi se obezbedilo ispravna simulacija funkcionisanje modela naplate putarine. Za skripte ulazak.sql i naplata.sql predvideti da će biti pokretane iz nekoliko sesija za svaku. Pojasniti potencijalne probleme u funkcionisanju simulacije ukoliko se ne upotrebljavaju transakcije (4 boda) i implementirati zahtevane mehanizme u skriptama koji će sprečiti te problem.

Napomena : Ne sme doći do situacije u kojoj imamo novi ulazak za vozilo koje je već u prometu, a da nije izašlo!