

U bazi podataka postoje sledeće tabele:

poste(sifra,naziv)

klijenti(sifra,naziv,sifraPoste)

paketi(sifra,sifraPosiljaoca,sifraPrimaoca,datumSlanja,željeniDatumPrimanja,realizovaniDatumPrimanja, naplata)

transferi(sifraPaketa,sifraPosteSlanja,sifraPostePrimanja,datumSlanja,datumPrimanja)

Svakom klijentu je pridružena jedna pošta. Paket jedinstveno određuje njegova šifra. Prilikom slanja paketa poznat je pošiljalac, primalac, datum slanja, željeni datum primanja i koliko je naplaćeno klijentu za uslugu slanja. Realizovani datum primanja je datum dospeća paketa u poštu klijenta primaoca. Kako paket može da putuje od pošte do pošte, svako slanje iz jedne i preuzimanje u drugoj predstavlja jedan transfer. Da bi paket stigao do pošte primaoca mora da postoji najmanje jedan transfer, čak i ako taj transfer beleži promet paketa direktno od pošte pošiljaoca do pošte primaoca. Za svaki transfer paketa poznati su pošta slanja, pošta primanja i datum slanja paketa.

ZADACI:

1. Data je storna procedura **sp_naplataPoKlijentu**. Ova procedura daje podatke o tome koliko je ukupno naplaćeno svakom klijentu za usluge slanja paketa. Podaci se prikazuju u formatu (sifraKlijenta, nazivKlijenta, naplaceno). Poznato je da je nad tabelom **klijenti** napravljen klasterovani primarni ključ nad kolonom **sifra**.

Koju od navedenih strategija indeksiranja tabele **paketi** treba odabrati pri optimizaciji izvršavanja date storne procedure:

- ⌚ Klasterovani primarni ključ nad kolonom **sifra** i neklasterovani indeks nad kolonom **sifraPosiljaoca**,
- ⌚ Neklasterovani primarni ključ nad kolonom **sifra** i klasterovani indeks nad kolonom **sifraPosiljaoca**?

Dati detaljno objašnjenje zašto je jedna strategija bolja od druge.

2. Date su tri storne procedure:

- ⌚ **sp_primanjePaketa** – Kada se transfer jednog paketa završi, procedura ažurira datum primanja u tom transferu, umesto NULL se upisuje trenutni datum;
- ⌚ **sp_poslatiPaketiKlijenta** – Za datu šifru klijenta, procedura prikazuje podatke o svim paketima koje je taj klijent poslao. Podaci se prikazuju u formatu (nazivPrimaoca, primljeno, trenutnaPosta) gde je **nazivPrimaoca** naziv klijenta koji prima paket; **primljeno** uzima vrednost “primljeno” ako je paket isporučen primaocu, ili “nije primljeno” u suprotnom; **trenutnaPosta** daje naziv poslednje pošte na koju je paket za sada stigao (pošta primanja u poslednjem dosadašnjem transferu paketa).
- ⌚ **sp_transferiPaketa** – Za datu šifru paketa, procedura prikazuje podatke o svim transferima tog paketa u formatu (nazivPosteSlanja, nazivPostePrimanja, datumSlanja, datumPrimanja). U slučaju da je transfer u toku, u kolonu **datumPrimanja** se upisuje “nije primljeno”. Transferi se prikazuju redom, od transfera iz pošte pošiljaoca do poslednjeg dosadašnjeg transfera paketa.

Postaviti primarne i spoljašnje ključeve nad tabelama. Ubrzati izvršavanje zadatih upita kreiranjem odgovarajućih indeksa tako da se postigne što manje zbirno vreme izvršavanja. Voditi računa da u svakoj tabeli količina memorije upotrebljena za indekse bude što manja u odnosu na memoriju upotrebljenu za podatke. Efikasnost upotrebe memorije donosi dodatne bodove. Neracionalna upotreba memorije ne donosi bodove.

Predato rešenje treba da sadrži:

- ⌚ SQL skriptu sa komandama za kreiranje potrebnih indeksa nad tabelama, zajedno sa primarnim i spoljašnjim ključevima;
- ⌚ Planove izvršavanja svih upita pre i posle upotrebe indeksa (za svaku proceduru/upit ponaosob). Za svaki plan izvršavanja napisati i 2-3 rečenice koje treba da pojašne dobijeno ubrzanje.

Napomena: Vreme rada je 60 minuta.