

2. VJEŽBA

Uvodne napomene:

Upiti (SQL naredbe) koji će se koristiti u ovoj vježbi ne moraju biti logički ispravni, odnosno dopušteni su i logički besmisleni upiti, poput npr. `SELECT * FROM salesOrder WHERE status > salesOrderID.`

Premda je poznato da najveći trošak izvršavanja upita proizlazi iz obavljanja UI operacija, u ovim vježbama će se kao okvirna mjera troška koristiti CPU vrijeme jer ga je vrlo jednostavno izmjeriti tijekom izvršavanja upita.

1. zadatak: Kreirati testnu bazu podataka labprof2

Procjena potrebnog vremena: 5 minuta
U rješenjima je potrebno dostaviti: ništa

Preuzeti datoteke *.csv i izvršiti skripta labprof2.sql.

U laboratoriju se očekuju sljedeća znanja i vještine:

- uništiti i ponovo kreirati bazu podataka labprof2

Relacije i podatke u relacijama koristiti za rješavanje zadataka u ovoj vježbi. Po potrebi, ako se podaci ili struktura relacija "pokvare" pri rješavanju zadataka, slobodno je ponovo kreirati i napuniti relacije ili ponovo kreirati bazu podataka labprof2.

2. zadatak: Utjecaj indeksa na operacije INSERT i DELETE

Procjena potrebnog vremena: 30 minuta
U rješenjima je potrebno dostaviti: SQL upite i njihov trošak izražen u CPU vremenu.

Eksperimentom utvrditi trošak operacije INSERT pri punjenju relacije velikom količinom podataka u slučaju kada nad relacijom nije kreiran indeks, u slučaju kada su kreirana dva indeksa i u slučaju kada su kreirana četiri indeksa. Slično, utvrditi trošak kada se iz relacije s velikom količinom podataka briše (DELETE) veliki broj n-torki: u slučaju kada nad relacijom nije kreiran indeks, kada su kreirana dva indeksa i kada su kreirana četiri indeksa.

Smjernice:

- trošak upita izražen u vremenu (CPU time) može se u alatu SSMS dobiti izvršavanjem SQL naredbi na sljedeći način:

```
SET STATISTICS TIME ON;
```


SQL naredba čije se CPU vrijeme mjeri; ispisat će se vrijeme izvršavanja ove naredbe

```
SET STATISTICS TIME OFF;
```


Napomena: Treba promatrati vrijeme ispisano ispod naslova "SQL Server Execution Times", jer u ovom kontekstu vrijeme utrošeno na "SQL Server parse and compile" nije važno.
- eksperiment može započeti npr. ovako: kreirati neku novu ciljnu relaciju bez indeksa i izmjeriti vrijeme potrebno za punjenje velikom količinom podataka (npr. više od 100 000 n-torki, izvršavanjem jedne INSERT naredbe). Kao izvor podataka može se koristiti postojeća relacija koja sadrži veliki broj n-torki, a kao ciljna relacija neka nova relacija sa strukturom jednakom postojećoj relaciji.

Tijekom nadziranih provjera/vježbi očekuju se sljedeća znanja i vještine:

- objasniti svoje eksperimente i njihove rezultate
- na zahtjev provesti slične eksperimente

3. zadatak: Utjecaj indeksa na operaciju UPDATE

Procjena potrebnog vremena: 20 minuta

U rješenjima je potrebno dostaviti: SQL upite i njihov trošak izražen u CPU vremenu, kratko obrazloženje rezultata (jednu ili dvije rečenice).

Eksperimentom utvrditi da isti indeks izgrađen nad istom relacijom koja sadrži iste podatke, za jednu UPDATE naredbu može biti koristan (vrijeme izvršavanja se skraćuje uz indeks u odnosu na vrijeme izvršavanja bez indeksa), a za drugu UPDATE naredbu može biti štetan (vrijeme izvršavanja se produljuje uz indeks u odnosu na vrijeme izvršavanja bez indeksa).

Smjernice:

- u eksperimentu koristiti UPDATE naredbe koje obavljaju izmjene nad velikim brojem n-torki

Tijekom nadziranih provjera/vježbi očekuju se sljedeća znanja i vještine:

- objasniti svoje eksperimente i njihove rezultate
- na zahtjev provesti slične eksperimente

4. zadatak: Faktor popunjenosti indeksa (*Fill factor*)

Procjena potrebnog vremena: 30 minuta

U rješenjima je potrebno dostaviti: obavljene SQL upite, podatke o dubini indeksa, utrošku prostora za indeks izražen u stranicama (pages) i trošak izvršavanja INSERT naredbi izražen u CPU vremenu.

- a) Relaciju bez indeksa napuniti velikom količinom podataka (npr. više od 100 000 n-torki). Kreirati indeks s vrlo visokim faktorom popunjenosti (100%). Zabilježiti dubinu indeksa i utrošak prostora za fizičku pohranu tog indeksa. Zatim u relaciju pomoću jedne INSERT naredbe dodati ponovo veliku količinu podataka (npr. više od 100 000 n-torki). Zabilježiti dubinu indeksa i utrošak prostora za fizičku pohranu tog indeksa, te vrijeme izvršavanja dotične naredbe INSERT.
- b) Ponoviti eksperiment pod a), ali uz početni faktor popunjenosti indeksa od 10%.

Tijekom nadziranih provjera/vježbi očekuju se sljedeća znanja i vještine:

- objasniti svoje eksperimente i njihove rezultate
- na zahtjev provesti slične eksperimente
- pročitati dubinu stabla za neki indeks koji je kreiran u bazi, objasniti pojam dubine stabla
- pročitati faktor popunjenosti za neki indeks koji je kreiran u bazi, objasniti pojam faktora popunjenosti
- odgovoriti na pitanje: kolika je veličina stranice (page) u sustavu SQL Server? Gdje se može pronaći taj podatak?

5. zadatak: *Clustered* indeks

Procjena potrebnog vremena: 25 minuta

U rješenjima je potrebno dostaviti: SQL upite, kratko obrazloženje i odgovor na pitanje navedeno uz zadatak.

Primjerom prikazati da veličina (broj utrošenih blokova) *non-clustered* indeksa koji je izgrađen nad nekom relacijom može znatno porasti u trenutku kad se nad istom relacijom kreira neki *clustered* indeks. Zašto izgradnja *clustered* indeksa povećava prostor kojeg zauzima postojeći *non-clustered* indeks kreiran nad istom relacijom?

Tijekom nadziranih provjera/vježbi očekuju se sljedeća znanja i vještine:

- objasniti svoj eksperiment i rezultate, objasniti svojstva, prednosti i nedostatke *clustered* indeksa
- na zahtjev provesti slične eksperimente