

OBLIKOVANJE BAZA PODATAKA

Vježba 7

T-SQL za kreiranje indeksa

- Izrada klasteriranog indeksa:

- Implicitno, definiranjem primarnog ključa ili eksplicitno:

```
CREATE CLUSTERED INDEX naziv ON tablica(stupac1, ...)
```

- Izrada neklasteriranog indeksa:

```
CREATE NONCLUSTERED INDEX naziv ON tablica(stupac1,...)
```

- Izrada neklasteriranog pokrivajućeg indeksa:

```
CREATE NONCLUSTERED INDEX naziv ON tablica(stupac1,...) INCLUDE (stupac1, ...)
```

- ALTER INDEX ne može promijeniti definiciju indeksa (iznimka)!

- Uklanjanje oba tipa indeksa:

```
DROP INDEX indeks ON tablica
```

Optimizacija upita pomoću indeksa

1. Uključiti **SET STATISTICS IO ON**
2. Izvršiti upit i zabilježiti broj pretraženih stranica (engl. *logical reads*)
3. **Algoritam optimizacije:**
 - a. Osigurati da svaka tablica ima klasterirani indeks
 - b. Identificirati sve stupce koji sudjeluju u WHERE dijelu upitu i jednog od njih izabrati za ključ neklasteriranog indeksa
 - c. Sve ostale stupce koji su dio upita uključiti u indeks s **INCLUDE** (neklasterirani indeks u sebi automatski sadrži vrijednost klasteriranog indeksa – ne treba ga ponovno dodavati)
4. Izvršiti upit i vidjeti je li došlo do poboljšanja (manjeg broja *logical readova*)

Vježba

- Uklonite sve indekse osim klasteriranih s tablica Proizvod i Racun. Postavite SET STATISTICS IO ON
1. Optimizirajte upit: `SELECT PotkategorijaID FROM Proizvod WHERE PotkategorijaID = 12`
 - a. Koliko stranica pregleda RDBMS?
 - b. Napravite indeks koji optimizira upit
 - c. Koliko sad stranica pregleda RDBMS?
 - d. Uklonite indeks.
 2. Optimizirajte upit: `SELECT IDProizvod, PotkategorijaID FROM Proizvod WHERE PotkategorijaID = 12`

Vježba

3. Optimizirajte upit:

```
SELECT IDProizvod, Naziv, PotkategorijaID FROM Proizvod  
WHERE PotkategorijaID = 12
```

Pomaže li nam indeks iz prethodnog primjera? Što učiniti sa stupcem Naziv?

4. Optimizirajte upit:

```
SELECT IDProizvod, Naziv, PotkategorijaID FROM Proizvod  
PotkategorijaID = 12 AND Naziv LIKE 'ML%'
```

5. Optimizirajte upit:

```
SELECT Boja, COUNT(*) AS Cnt FROM Proizvod  
WHERE PotkategorijaID = 12 GROUP BY Boja ORDER BY Cnt DESC
```

Vježba

6. Optimizirajte upit:

```
SELECT DatumIzdavanja FROM Racun  
WHERE DatumIzdavanja BETWEEN '20010702' AND '20010702 23:59:59'
```

- a. Koliko stranica pregleda RDBMS?
- b. Napravite indeks koji optimizira upit
- c. Koliko sad stranica pregleda RDBMS?
- d. Uklonite indeks.

7. Optimizirajte upit:

```
SELECT IDRacun, DatumIzdavanja FROM Racun  
WHERE DatumIzdavanja BETWEEN '20010702' AND '20010702 23:59:59'
```

Vježba

8. Optimizirajte upit:

```
SELECT IDRacun, BrojRacuna, DatumIzdavanja FROM Racun  
WHERE DatumIzdavanja BETWEEN '20010702' AND '20010702 23:59:59'
```

Pomaže li nam indeks iz prethodnog primjera?
Što učiniti sa stupcem BrojRacuna?

9. Optimizirajte upit:

```
SELECT IDRacun FROM Racun  
WHERE DatumIzdavanja BETWEEN '20010702' AND '20010702 23:59:59' AND BrojRacuna LIKE 'S%'
```

10. Optimizirajte upit:

```
SELECT KupacID, COUNT(*) AS Cnt FROM Racun  
WHERE DatumIzdavanja BETWEEN '20010702' AND '20010702 23:59:59'  
GROUP BY KupacID ORDER BY Cnt DESC
```