

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ
НАЦІОНАЛЬНОМУ УНІВЕРСИТЕТІ “ЛЬВІВСЬКА ПОЛІТЕХНІКА”



Лабораторна робота

з дисципліни
«ОБДЗ»

Виконав:
студент групи КН-209
Мацкула Олег
Викладач:
Мельникова Н. І.

Львів – 2020 р.

Лабораторна робота №13

Аналіз та оптимізація запитів

Мета: Навчитися аналізувати роботу СУБД та оптимізовувати виконання складних запитів на вибірку даних. Виконати аналіз складних запитів за допомогою директиви EXPLAIN, модифікувати найповільніші запити з метою їх пришвидчення.

Хід роботи

1. Для початку проаналізуємо запит.

```
EXPLAIN ANALYSE SELECT DISTINCT sf.name
FROM healthy_core_supplement s
     INNER JOIN healthy_core_supplement_side_effects ssf on s.id = ssf.supplement_id
     INNER JOIN healthy_core_sideeffect sf on ssf.sideeffect_id = sf.id
WHERE s.name = 'Sugar'
```

```
Unique (cost=9.65..9.68 rows=5 width=274) (actual time=0.151..0.154 rows=4 loops=1)
-> Sort (cost=9.65..9.67 rows=5 width=274) (actual time=0.150..0.151 rows=4 loops=1)
    Sort Key: sf.name
    Sort Method: quicksort Memory: 25kB
-> Nested Loop (cost=2.01..9.59 rows=5 width=274) (actual time=0.059..0.138 rows=4 loops=1)
    -> Hash Join (cost=1.86..8.04 rows=5 width=4) (actual time=0.045..0.117 rows=4 loops=1)
        Hash Cond: (ssf.supplement_id = s.id)
        -> Seq Scan on healthy_core_supplement_side_effects ssf (cost=0.00..5.28 rows=328 width=8) (actual time=0.015..0.045 rows=210 loops=1)
        -> Hash (cost=1.85..1.85 rows=1 width=4) (actual time=0.023..0.023 rows=1 loops=1)
            Buckets: 1024 Batches: 1 Memory Usage: 9kB
            -> Seq Scan on healthy_core_supplement s (cost=0.00..1.85 rows=1 width=4) (actual time=0.013..0.020 rows=1 loops=1)
                Filter: ((name)::text = 'Sugar'::text)
                Rows Removed by Filter: 67
    -> Index Scan using healthy_core_sideeffect_pkey on healthy_core_sideeffect sf (cost=0.15..0.31 rows=1 width=278) (actual time=0.004..0.004 rows=1 loops=1)
        Index Cond: (id = ssf.sideeffect_id)
Planning Time: 0.359 ms
Execution Time: 0.201 ms
```

Сумарний час виконання запиту – 0.56 ms

2. Спробуємо додати індекс на ім'я у таблиці healthy_core_supplement

```
create index healthy_core_supplement_name_index
on healthy_core_supplement (name);
```

3. Перевіримо виконання запиту ще раз

```
Planning Time: 0.357 ms
Execution Time: 0.123 ms
```

Оптимізація спрацювала.

Примітка: Насправді така зміна часу викликана більше високою дисперсією часу виконання. На малих даних оптимізації не є дуже суттєвими. Однак на великих базах така оптимізація зможе зекономити більше половини нашого з вами часу.

Висновок: Отримав теоритичні знання з оптимізації SQL запитів. Використав директиву EXPLAIN для аналізу виконання складного запиту.