МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ
«ЛЬВІВСЬКА ПОЛІТЕХНІКА»

Інститут ІКНІ

Кафедра Систем Штучного Інтелекту

Організація баз даних та знань

Лабораторна робота

Виконав:

Кіндрат В.Р.

КН-209

Викладач:

Мельникова Н. І.

Львів – 2020

Лабораторна робота №13

з курсу “ОБДЗ”

на тему: “Аналіз та оптимізація запитів”

Мета роботи: 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.

Короткі теоретичні відомості.

Для аналізу виконання запитів в MySQL існує декілька спеціальних директив. Основна з них – EXPLAIN. Директива EXPLAIN дозволяє визначити поля таблиці, для яких варто створити додаткові індекси, щоб пришвидшити вибірку даних. Індекс – це механізм, який підвищує швидкість пошуку та доступу до записів за індексованими полями. Загалом, варто створювати індекси для тих полів, за якими відбувається з’єднання таблиць, перевірка умови чи пошук. За допомогою директиви EXPLAIN також можна визначити послідовність, в якій відбувається з’єднання таблиць при вибірці даних. Якщо оптимізатор вибирає не найкращу послідовність з’єднання таблиць, потрібно використати опцію STRAIGHT_JOIN директиви SELECT. Тоді з’єднання таблиць буде відбуватись в тому порядку, в якому перераховані таблиці у запиті. Також, за допомогою опцій FORCE INDEX, USE INDEX та IGNORE INDEX можна керувати використанням індексів у випадку їх неправильного вибору оптимізатором, тобто, якщо вони не підвищують ефективність вибірки рядків.

Завдання на лабораторну роботу

1. Визначити індекси таблиці.

2. Створити додаткові індекси для таблиці.

3. Дослідити процес виконання запитів за допомогою EXPLAIN.

Хід роботи

1. За допомогою директиви SHOW INDEX визначаю наявні індекси для таблиць client та worker

SHOW INDEX FROM client;

```
mysql> SHOW INDEX FROM client;
+--------+------------+--------------+--------------+--------------+-----------+-------------+----------+--------+------+------------+---------+---------------+---------+------------+
| Table  | Non_unique | Key_name     | Seq_in_index | Column_name  | Collation | Cardinality | Sub_part | Packed | Null | Index_type | Comment | Index_comment | Visible | Expression |
+--------+------------+--------------+--------------+--------------+-----------+-------------+----------+--------+------+------------+---------+---------------+---------+------------+
| client |          0 | PRIMARY      |            1 | id           | A         |           3 |     NULL |   NULL |      | BTREE      |         |               | YES     | NULL       |
| client |          0 | phone_number |            1 | phone_number | A         |           3 |     NULL |   NULL | YES  | BTREE      |         |               | YES     | NULL       |
+--------+------------+--------------+--------------+--------------+-----------+-------------+----------+--------+------+------------+---------+---------------+---------+------------+
2 rows in set (0.00 sec)
```

SHOW INDEX FROM worker;

```
mysql> SHOW INDEX from worker;
+--------+------------+----------+--------------+-------------+-----------+-------------+----------+--------+------+------------+---------+---------------+---------+------------+
| Table  | Non_unique | Key_name | Seq_in_index | Column_name | Collation | Cardinality | Sub_part | Packed | Null | Index_type | Comment | Index_comment | Visible | Expression |
+--------+------------+----------+--------------+-------------+-----------+-------------+----------+--------+------+------------+---------+---------------+---------+------------+
| worker |          0 | PRIMARY  |            1 | id          | A         |           1 |     NULL |   NULL |      | BTREE      |         |               | YES     | NULL       |
+--------+------------+----------+--------------+-------------+-----------+-------------+----------+--------+------+------------+---------+---------------+---------+------------+
1 row in set (0.11 sec)
```

2. Створюю новий індекс в таблиці client. Так як для пошуку інформації часто використовується прізвище та ім’я.

CREATE INDEX clientINDX3 ON client (first_name, second_name);

```
mysql> CREATE INDEX clientINDX3 ON client (first_name, second_name );
Query OK, 0 rows affected (0.45 sec)
Records: 0  Duplicates: 0  Warnings: 0
```

```
mysql> SHOW INDEX FROM client;
+--------+------------+---------------+--------------+-------------+-----------+-------------+----------+--------+------+------------+---------+---------------+---------+------------+
| Table  | Non_unique | Key_name      | Seq_in_index | Column_name | Collation | Cardinality | Sub_part | Packed | Null | Index_type | Comment | Index_comment | Visible | Expression |
+--------+------------+---------------+--------------+-------------+-----------+-------------+----------+--------+------+------------+---------+---------------+---------+------------+
| client |          0 | PRIMARY       |            1 | id          | A         |           3 |     NULL |   NULL |      | BTREE      |         |               | YES     | NULL       |
| client |          0 | phone_number  |            1 | phone_number| A         |           3 |     NULL |   NULL | YES  | BTREE      |         |               | YES     | NULL       |
| client |          1 | clientINDX3   |            1 | first_name  | A         |           2 |     NULL |   NULL |      | BTREE      |         |               | YES     | NULL       |
| client |          1 | clientINDX3   |            2 | second_name | A         |           2 |     NULL |   NULL |      | BTREE      |         |               | YES     | NULL       |
+--------+------------+---------------+--------------+-------------+-----------+-------------+----------+--------+------+------------+---------+---------------+---------+------------+
4 rows in set (0.08 sec)
```

Створюю індекс в таблиці worker, для пошуку працівників по виду діяльності (post).

CREATE INDEX postINDX ON worker(post);

```
mysql> CREATE UNIQUE INDEX postINDX ON worker(post);
Query OK, 0 rows affected (0.53 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> SHOW INDEX from worker;
+--------+------------+----------+--------------+-------------+-----------+-------------+----------+--------+------+------------+---------+---------------+---------+------------+
| Table  | Non_unique | Key_name | Seq_in_index | Column_name | Collation | Cardinality | Sub_part | Packed | Null | Index_type | Comment | Index_comment | Visible | Expression |
+--------+------------+----------+--------------+-------------+-----------+-------------+----------+--------+------+------------+---------+---------------+---------+------------+
| worker |          0 | PRIMARY  |            1 | id          | A         |           1 |     NULL |   NULL |      | BTREE      |         |               | YES     | NULL       |
| worker |          0 | postINDX |            1 | post        | A         |           1 |     NULL |   NULL |      | BTREE      |         |               | YES     | NULL       |
+--------+------------+----------+--------------+-------------+-----------+-------------+----------+--------+------+------------+---------+---------------+---------+------------+
2 rows in set (0.03 sec)
```

3. Виконаю аналіз виконання запиту використовуючи EXPLAIN та опцію STRAIGHT_JOIN.

EXPLAIN SELECT eviction_date FROM `order` INNER JOIN client

WHERE client.first_name = 'Vova'

AND client.second_name = 'Korzan';

```
mysql> EXPLAIN SELECT eviction_date FROM `order` INNER JOIN client WHERE  client.first_name = 'Vova' AND client.second_name = 'Korzan';
+----+-------------+--------+------------+------+---------------+-------------+---------+-------------+------+----------+----------------------------------------------+
| id | select_type | table  | partitions | type | possible_keys | key         | key_len | ref         | rows | filtered | Extra                                        |
+----+-------------+--------+------------+------+---------------+-------------+---------+-------------+------+----------+----------------------------------------------+
|  1 | SIMPLE      | client | NULL       | ref  | clientINDX3   | clientINDX3 | 204     | const,const |    1 |   100.00 | Using index                                  |
|  1 | SIMPLE      | order  | NULL       | ALL  | NULL          | NULL        | NULL    | NULL        |    3 |   100.00 | Using join buffer (Block Nested Loop)        |
+----+-------------+--------+------------+------+---------------+-------------+---------+-------------+------+----------+----------------------------------------------+
2 rows in set, 1 warning (0.01 sec)
```

EXPLAIN SELECT STRAIGHT_JOIN first_name AS name

FROM worker INNER JOIN room_worker

WHERE worker.post = 'cleaner'

AND room_worker.room_id > 2;

```
mysql> EXPLAIN SELECT STRAIGHT_JOIN first_name AS name FROM worker INNER JOIN room_worker
    -> WHERE worker.post = 'cleaner' AND room_worker.room_id > 2;
+----+-------------+-------------+------------+-------+---------------+-------------+---------+-------+------+----------+--------------------------+
| id | select_type | table       | partitions | type  | possible_keys | key         | key_len | ref   | rows | filtered | Extra                    |
+----+-------------+-------------+------------+-------+---------------+-------------+---------+-------+------+----------+--------------------------+
|  1 | SIMPLE      | worker      | NULL       | const | postINDX      | postINDX    | 102     | const |    1 |   100.00 | NULL                     |
|  1 | SIMPLE      | room_worker | NULL       | range | Room_worker0  | Room_worker0 | 4       | NULL  |    3 |   100.00 | Using where; Using index |
+----+-------------+-------------+------------+-------+---------------+-------------+---------+-------+------+----------+--------------------------+
2 rows in set, 1 warning (0.00 sec)
```

Висновок

Під час виконання цієї лабораторної роботи я навчився аналізувати запити директивою EXPLAIN та оптимізувати виконання запитів модифікацією порядку з’єднань таблиць і створення додаткових індексів.