**МІНІСТЕРСТВО ОСВІТИ І НАУКИ**
**НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «ЛЬВІВСЬКА ПОЛІТЕХНІКА»**
**Інститут комп'ютерних наук та інформаційних технологій**
**Кафедра систем штучного інтелекту**


# Звіт до практичної роботи №13

**з дисципліни “ ОБДЗ ”**

**На тему: «Аналіз та оптимізація запитів»**

**Виконав**:
ст. гр. КН-211
Качмарик Віктор
**Викладач:**
Якимишин Х. М.

Львів – 2020

**Мета роботи:** Навчитися використовувати механізм транзакцій у СУБД MySQL. Розробити SQL запити, які виконуються як єдине ціле в рамках однієї транзакції.

## 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.

Опис директив.

SELECT BENCHMARK(кількість_циклів, вираз)

Виконує вираз вказану кількість разів, і повертає загальний час виконання.

EXPLAIN SELECT …

Використовується разом із запитом SELECT. Виводить інформацію про план обробки і виконання запиту, включно з інформацією про те, як і в якому порядку з'єднувались таблиці. EXPLAIN EXTENDED виводить розширену інформацію.

Результати директиви виводяться у вигляді рядків з такими полями:

id – порядковий номер директиви SELECT у запиті;

select_type – тип вибірки (simple, primary, union, subquery, derived, uncachable subquery тощо);

table – назва таблиці, для якої виводиться інформація;
type – тип з’єднання (system, const, eq_ref, ref, fulltext, range тощо);
possible_keys – індекси, які наявні у таблиці, і можуть бути використані;
key – назва індексу, який було обрано для виконання запиту;
key_len – довжина індекса, який був використаний при виконанні запиту;
ref – вказує, які рядки чи константи порівнюються зі значенням індекса при відборі;
rows – (прогнозована) кількість рядків, потрібних для виконання запиту;
Extra – додаткові дані про хід виконання запиту.

ANALYZE TABLE

Оновлює статистичну інформацію про таблицю (наприклад, поточний розмір ключових полів). Ця інформація впливає на роботу оптимізатора запитів, і може вплинути на вибір індексів при виконанні запитів.

SHOW INDEX FROM ім’я_таблиці

Виводить інформацію про індекси таблиці.

CREATE [UNIQUE | FULLTEXT] INDEX назва
ON ім’я_таблиці (перелік_полів)

Створює індекс для одного або декількох полів таблиці. Одне поле може входити до кількох індексів. Якщо індекс оголошено як UNIQUE, то значення відповідних полів таблиці повинні бути унікальними. Таблиці MyISAM підтримують створення повнотекстових індексів (FULLTEXT) для полів типу TEXT, CHAR, VARCHAR.

## Завдання:

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 визначимо наявні індекси для таблиць order, customer та order_item.

```
mysql> show index from `order`;
+-------+------------+----------+--------------+-------------+-----------+-------------+----------+--------+------+------------+---------+---------------+---------+------------+
| Table | Non_unique | Key_name | Seq_in_index | Column_name | Collation | Cardinality | Sub_part | Packed | Null | Index_type | Comment | Index_comment | Visible | Expression |
+-------+------------+----------+--------------+-------------+-----------+-------------+----------+--------+------+------------+---------+---------------+---------+------------+
| order |          0 | PRIMARY  |            1 | order_id    | A         |          15 |     NULL |   NULL |      | BTREE      |         |               | YES     | NULL       |
| order |          1 | order_fk0 |            1 | customer_id | A         |           9 |     NULL |   NULL |      | BTREE      |         |               | YES     | NULL       |
| order |          1 | order_fk1 |            1 | pharmacy_id | A         |           8 |     NULL |   NULL | YES  | BTREE      |         |               | YES     | NULL       |
+-------+------------+----------+--------------+-------------+-----------+-------------+----------+--------+------+------------+---------+---------------+---------+------------+
3 rows in set (0.01 sec)

mysql> show index from customer;
+----------+------------+----------+--------------+-------------+-----------+-------------+----------+--------+------+------------+---------+---------------+---------+------------+
| Table    | Non_unique | Key_name | Seq_in_index | Column_name | Collation | Cardinality | Sub_part | Packed | Null | Index_type | Comment | Index_comment | Visible | Expression |
+----------+------------+----------+--------------+-------------+-----------+-------------+----------+--------+------+------------+---------+---------------+---------+------------+
| customer |          0 | PRIMARY  |            1 | customer_id | A         |          10 |     NULL |   NULL |      | BTREE      |         |               | YES     | NULL       |
+----------+------------+----------+--------------+-------------+-----------+-------------+----------+--------+------+------------+---------+---------------+---------+------------+
1 row in set (0.00 sec)

mysql> show index from order_item;
+------------+------------+---------------+--------------+-------------+-----------+-------------+----------+--------+------+------------+---------+---------------+---------+------------+
| Table      | Non_unique | Key_name      | Seq_in_index | Column_name | Collation | Cardinality | Sub_part | Packed | Null | Index_type | Comment | Index_comment | Visible | Expression |
+------------+------------+---------------+--------------+-------------+-----------+-------------+----------+--------+------+------------+---------+---------------+---------+------------+
| order_item |          0 | PRIMARY       |            1 | item_id     | A         |          24 |     NULL |   NULL |      | BTREE      |         |               | YES     | NULL       |
| order_item |          1 | order_item_fk0 |            1 | medicine_id | A         |          14 |     NULL |   NULL |      | BTREE      |         |               | YES     | NULL       |
| order_item |          1 | order_item_fk1 |            1 | order_id    | A         |          15 |     NULL |   NULL |      | BTREE      |         |               | YES     | NULL       |
+------------+------------+---------------+--------------+-------------+-----------+-------------+----------+--------+------+------------+---------+---------------+---------+------------+
3 rows in set (0.01 sec)
```

2. Створимо новий індекс для таблиці order, customer та order_item. У БД є декілька запитів, які здійснюють вибірку даних за логіном покупця (поле login), за id лік(medicine_id). Створення індексів для цих полів повинно оптимізувати виконання запитів.

```
mysql> CREATE INDEX customerNewIndex ON customer (customer_id, login, first_name);
Query OK, 0 rows affected (0.04 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> SHOW INDEX FROM customer;
+----------+------------+------------------+--------------+-------------+-----------+-------------+----------+--------+------+------------+---------+---------------+---------+------------+
| Table    | Non_unique | Key_name         | Seq_in_index | Column_name | Collation | Cardinality | Sub_part | Packed | Null | Index_type | Comment | Index_comment | Visible | Expression |
+----------+------------+------------------+--------------+-------------+-----------+-------------+----------+--------+------+------------+---------+---------------+---------+------------+
| customer |          0 | PRIMARY          |            1 | customer_id | A         |          10 |     NULL |   NULL |      | BTREE      |         |               | YES     | NULL       |
| customer |          1 | customerNewIndex |            1 | customer_id | A         |          10 |     NULL |   NULL |      | BTREE      |         |               | YES     | NULL       |
| customer |          1 | customerNewIndex |            2 | login       | A         |          10 |     NULL |   NULL |      | BTREE      |         |               | YES     | NULL       |
| customer |          1 | customerNewIndex |            3 | first_name  | A         |          10 |     NULL |   NULL |      | BTREE      |         |               | YES     | NULL       |
+----------+------------+------------------+--------------+-------------+-----------+-------------+----------+--------+------+------------+---------+---------------+---------+------------+


mysql> SHOW INDEX FROM `order`;
+-------+------------+---------------+--------------+-------------+-----------+-------------+----------+--------+------+------------+---------+---------------+---------+------------+
| Table | Non_unique | Key_name      | Seq_in_index | Column_name | Collation | Cardinality | Sub_part | Packed | Null | Index_type | Comment | Index_comment | Visible | Expression |
+-------+------------+---------------+--------------+-------------+-----------+-------------+----------+--------+------+------------+---------+---------------+---------+------------+
| order |          0 | PRIMARY       |            1 | order_id    | A         |          15 |     NULL |   NULL |      | BTREE      |         |               | YES     | NULL       |
| order |          1 | order_fk1     |            1 | pharmacy_id | A         |           8 |     NULL |   NULL | YES  | BTREE      |         |               | YES     | NULL       |
| order |          1 | orderNewIndex |            1 | customer_id | A         |           9 |     NULL |   NULL |      | BTREE      |         |               | YES     | NULL       |
| order |          1 | orderNewIndex |            2 | order_id    | A         |          15 |     NULL |   NULL |      | BTREE      |         |               | YES     | NULL       |
+-------+------------+---------------+--------------+-------------+-----------+-------------+----------+--------+------+------------+---------+---------------+---------+------------+


mysql> CREATE INDEX order_itemNewIndex ON order_item (order_id, medicine_id);
Query OK, 0 rows affected (0.04 sec)
Records: 0  Duplicates: 0  Warnings: 0

mysql> SHOW INDEX FROM order_item;
+------------+------------+-------------------+--------------+-------------+-----------+-------------+----------+--------+------+------------+---------+---------------+---------+------------+
| Table      | Non_unique | Key_name          | Seq_in_index | Column_name | Collation | Cardinality | Sub_part | Packed | Null | Index_type | Comment | Index_comment | Visible | Expression |
+------------+------------+-------------------+--------------+-------------+-----------+-------------+----------+--------+------+------------+---------+---------------+---------+------------+
| order_item |          0 | PRIMARY           |            1 | item_id     | A         |          24 |     NULL |   NULL |      | BTREE      |         |               | YES     | NULL       |
| order_item |          1 | order_item_fk0    |            1 | medicine_id | A         |          14 |     NULL |   NULL |      | BTREE      |         |               | YES     | NULL       |
| order_item |          1 | order_itemNewIndex |            1 | order_id    | A         |          15 |     NULL |   NULL |      | BTREE      |         |               | YES     | NULL       |
| order_item |          1 | order_itemNewIndex |            2 | medicine_id | A         |          24 |     NULL |   NULL |      | BTREE      |         |               | YES     | NULL       |
+------------+------------+-------------------+--------------+-------------+-----------+-------------+----------+--------+------+------------+---------+---------------+---------+------------+
```

3. Виконаємо аналіз виконання складного запиту з однієї з попередніх робіт використовуючи EXPLAIN та опцію STRAIGHT_JOIN:

```
mysql> EXPLAIN SELECT customer.first_name AS customer, customer.login,
    -> AVG(order_item.quantity*medicine.price) AS AverCosts
    -> FROM ((order_item INNER JOIN medicine) INNER JOIN customer) INNER JOIN `order`
    -> ON customer.customer_id = `order`.customer_id
    -> AND `order`.order_id = order_item.order_id
    -> AND order_item.medicine_id = medicine.medicine_id;
+----+-------------+------------+------------+-------+-------------------------------+-------------------+---------+-------------------------------+------+----------+--------------------------+
| id | select_type | table      | partitions | type  | possible_keys                 | key               | key_len | ref                           | rows | filtered | Extra                    |
+----+-------------+------------+------------+-------+-------------------------------+-------------------+---------+-------------------------------+------+----------+--------------------------+
|  1 | SIMPLE      | customer   | NULL       | index | PRIMARY,customerNewIndex      | customerNewIndex  | 768     | NULL                          |   10 |   100.00 | Using where; Using index |
|  1 | SIMPLE      | order      | NULL       | ref   | PRIMARY,orderNewIndex         | orderNewIndex     | 4       | pharmacy.customer.customer_id |    1 |   100.00 | Using index              |
|  1 | SIMPLE      | order_item | NULL       | ref   | order_item_fk0,order_itemNewIndex | order_itemNewIndex | 4       | pharmacy.order.order_id       |    1 |   100.00 | NULL                     |
|  1 | SIMPLE      | medicine   | NULL       | eq_ref | PRIMARY                      | PRIMARY           | 4       | pharmacy.order_item.medicine_id |    1 |   100.00 | NULL                     |
+----+-------------+------------+------------+-------+-------------------------------+-------------------+---------+-------------------------------+------+----------+--------------------------+
4 rows in set, 1 warning (0.00 sec)

mysql> EXPLAIN SELECT STRAIGHT_JOIN customer.first_name AS customer, customer.login,
    -> AVG(order_item.quantity*medicine.price) AS AverCosts
    -> FROM ((order_item INNER JOIN medicine) INNER JOIN customer) INNER JOIN `order`
    -> ON customer.customer_id = `order`.customer_id
    -> AND `order`.order_id = order_item.order_id
    -> AND order_item.medicine_id = medicine.medicine_id;
+----+-------------+------------+------------+--------+-------------------------------+-------------------+---------+-------------------------------+------+----------+-----------------------------------------------+
| id | select_type | table      | partitions | type   | possible_keys                 | key               | key_len | ref                           | rows | filtered | Extra                                         |
+----+-------------+------------+------------+--------+-------------------------------+-------------------+---------+-------------------------------+------+----------+-----------------------------------------------+
|  1 | SIMPLE      | order_item | NULL       | ALL    | order_item_fk0,order_itemNewIndex | NULL              | NULL    | NULL                          |   24 |   100.00 | NULL                                          |
|  1 | SIMPLE      | medicine   | NULL       | eq_ref | PRIMARY                       | PRIMARY           | 4       | pharmacy.order_item.medicine_id |    1 |   100.00 | NULL                                          |
|  1 | SIMPLE      | customer   | NULL       | index  | PRIMARY,customerNewIndex      | customerNewIndex  | 768     | NULL                          |   10 |   100.00 | Using index; Using join buffer (Block Nested Loop) |
|  1 | SIMPLE      | order      | NULL       | eq_ref | PRIMARY,orderNewIndex         | PRIMARY           | 4       | pharmacy.order_item.order_id  |    1 |    11.11 | Using where                                   |
+----+-------------+------------+------------+--------+-------------------------------+-------------------+---------+-------------------------------+------+----------+-----------------------------------------------+
4 rows in set, 1 warning (0.00 sec)
```

**Висновок:** на даній лабораторній роботі я навчився аналізувати і оптимізувати виконання запитів. Для аналізу запитів було використано директиву EXPLAIN, а для оптимізації – модифікація порядку з’єднання таблиць і створення додаткових індексів.