

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ НАЦІОНАЛЬНОМУ УНІВЕРСИТЕТУ “ЛЬВІВСЬКА ПОЛІТЕХНІКА”

Кафедра систем штучного інтелекту

Звіт

про виконання лабораторної роботи №13

«Аналіз та оптимізація запитів»

з дисципліни «ОБДЗ»

Виконав:

студент групи КН-209

Ханас Михайло-Юрій

Викладач:

Мельникова Н.І.

Львів – 2020 р.

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.

Хід роботи

1. Створимо 2 таблиці(performance, auditorium) та заповнимо їх.

```
CREATE DATABASE teatr;
USE teatr;
CREATE TABLE auditorium
(
    id      INT AUTO_INCREMENT PRIMARY KEY,
    name    VARCHAR(32) NOT NULL,
    seats_number INT NOT NULL
);

CREATE TABLE performance
(
    id          INT AUTO_INCREMENT PRIMARY KEY,
    name        VARCHAR(32) NOT NULL,
    price       VARCHAR(32) NOT NULL,
    duration    INT NOT NULL ,
    id_auditorium INT NOT NULL ,
    data_performance TIMESTAMP NOT NULL,
    CONSTRAINT fk_performance_auditorium
        FOREIGN KEY (id_auditorium)
            REFERENCES auditorium (id)
            ON DELETE CASCADE
);
```

```
INSERT INTO auditorium VALUES
(1,"Big auditorium", 240),
(2,"Small auditorium", 110),
(3,"Middle auditorium", 170);
```

```
INSERT INTO performance VALUES
(1, "Muna Mazailo", 200, 2, 1, "2020-03-04 10:10:00"),
(2, "Stolen happiness", 400, 3, 2, "2020-03-03 10:50:00"),
(3, "Coriolanus", 600, 3, 2, "2020-04-03 15:00:00"),
(4, "Fireplace master", 450, 2, 3, "2020-04-04 16:30:00");
```

2. За допомогою SHOW INDEX визначаю наявні індекси для таблиць auditorium та performance.

SHOW INDEX FROM performance;

Table	Non_unique	Key_name	Seq_in_index	Column_name	Collation	Cardinality	Sub_part	Packed	Null	Index_type	Comment	Index_comment
performance	0	PRIMARY	1	id	A	4	NULL	NULL		BTREE		
performance	1	fk_performance_auditorium	1	id_auditorium	A	4	NULL	NULL		BTREE		

SHOW INDEX FROM auditorium;

Table	Non_unique	Key_name	Seq_in_index	Column_name	Collation	Cardinality	Sub_part	Packed	Null	Index_type	Comment	Index_comment
auditorium	0	PRIMARY	1	id	A	3	NULL	NULL		BTREE		

3. Створюємо новий індекс в таблиці performance. Створюю індекс для name в таблиці performance.

```
CREATE INDEX performance_name ON performance(name);
```

Table	Non_unique	Key_name	Seq_in_index	Column_name	Collation	Cardinality	Sub_part	Packed	Null	Index_type	Comment	Index_comment
performance	0	PRIMARY	1	id	A	4	NULL	NULL		BTREE		
performance	1	fk_performance_auditorium	1	id_auditorium	A	4	NULL	NULL		BTREE		
performance	1	performance_name	1	name	A	4	NULL	NULL		BTREE		

4. Створюємо новий індекс в таблиці auditorium. Створюю індекс для name в таблиці auditorium.

```
CREATE INDEX seats_number ON auditorium(seats_number);
```

Table	Non_unique	Key_name	Seq_in_index	Column_name	Collation	Cardinality	Sub_part	Packed	Null	Index_type	Comment	Index_comment
auditorium	0	PRIMARY	1	id	A	3	NULL	NULL		BTREE		
auditorium	1	seats_number	1	seats_number	A	3	NULL	NULL		BTREE		

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

```
EXPLAIN SELECT *
```

```
FROM auditorium
```

```
INNER JOIN performance
```

```
WHERE performance.name="Stolen happiness";
```

id	select_type	table	type	possible_keys	key	key_len	ref	rows	Extra
1	SIMPLE	performance	ref	performance_name	performance_name	34	const	1	Using index condition
1	SIMPLE	auditorium	ALL	NULL	NULL	NULL	NULL	3	Using join buffer (flat, BNL join)

```
EXPLAIN SELECT straight_join
```

```
performance.name as performance_name
```

```
FROM performance INNER JOIN auditorium
```

```
WHERE auditorium.name="Big auditorium";
```

id	select_type	table	type	possible_keys	key	key_len	ref	rows	Extra
1	SIMPLE	performance	index	NULL	performance_name	34	NULL	4	Using index
1	SIMPLE	auditorium	ALL	NULL	NULL	NULL	NULL	3	Using where; Using join buffer (flat, BNL join)

Висновок: на лабораторній роботі я навчився 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.