

**МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
НАЦІОНАЛЬНОМУ УНІВЕРСИТЕТІ “ЛЬВІВСЬКА  
ПОЛІТЕХНІКА”**

**Кафедра систем штучного інтелекту**



**Лабораторна робота #13**

**на тему:**

**«Аналіз та оптимізація запитів»**

з дисципліни

«Організація баз даних та знань»

**Виконав:**

студент групи КН-210

Максим Романьчук

**Викладач:**

Мельникова Н.І.

Львів – 2020 р.

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

### Хід роботи:

1. Переглянути індекси таблиць Worker і Office.

SHOW INDEX FROM office;

	Table	Non_unique	Key_name	Seq_in_index	Column_name	Collation	Cardinality	Sub_part	Packed	Null	Index_type
►	office	0	PRIMARY	1	id	A	3	NULL	NULL		BTREE
	office	0	office_id_UNIQUE	1	id	A	3	NULL	NULL		BTREE
	office	0	UGREOU code_UNIQUE	1	UGREOU code	A	3	NULL	NULL		BTREE

SHOW INDEX FROM worker;

	Table	Non_unique	Key_name	Seq_in_index	Column_name	Collation	Cardinality	Sub_part	Packed	Null	Index_type
►	worker	0	PRIMARY	1	id	A	5	NULL	NULL		BTREE
	worker	0	id_UNIQUE	1	id	A	5	NULL	NULL		BTREE
	worker	1	fk_worker_office1_idx	1	office_id	A	3	NULL	NULL		BTREE

2. Створити нові індекси для ім'я та прізвища працівника та назви офісу.

CREATE UNIQUE INDEX worker\_name ON worker (`name`, surname);  
SHOW INDEX FROM worker;

	Table	Non_unique	Key_name	Seq_in_index	Column_name	Collation	Cardinality	Sub_part	Packed	Null	Index_type
►	worker	0	PRIMARY	1	id	A	5	NULL	NULL		BTREE
	worker	0	id_UNIQUE	1	id	A	5	NULL	NULL		BTREE
	worker	0	worker_name	1	name	A	7	NULL	NULL		BTREE
	worker	0	worker_name	2	surname	A	7	NULL	NULL		BTREE
	worker	1	fk_worker_office1_idx	1	office_id	A	3	NULL	NULL		BTREE

CREATE INDEX office\_name ON office (`name`);  
SHOW INDEX FROM office;

	Table	Non_unique	Key_name	Seq_in_index	Column_name	Collation	Cardinality	Sub_part	Packed	Null	Index_type
►	office	0	PRIMARY	1	id	A	3	NULL	NULL		BTREE
	office	0	office_id_UNIQUE	1	id	A	3	NULL	NULL		BTREE
	office	0	UGREOU code_UNIQUE	1	UGREOU code	A	3	NULL	NULL		BTREE
	office	1	office_name	1	name	A	3	NULL	NULL		BTREE

3. Дослідити процес виконання запитів за допомогою EXPLAIN та опції STRAIGHT\_JOIN на прикладі запиту з попередніх лабораторних.

```

EXPLAIN SELECT DISTINCT office.`name` AS office, dealer.`name` AS dealer
FROM mydb.goods
INNER JOIN mydb.office
    ON goods.office_id = office.id
INNER JOIN mydb.dealer
    ON goods.dealer_id = dealer.id
WHERE goods.price <= 3500;

```

	id	select_type	table	partitions	type	possible_keys	key	key_len	ref	rows	filtered
►	1	SIMPLE	goods	<small>NULL</small>	ALL	fk_goods_dealer1_idx, fk_goods_offic...	<small>NULL</small>	<small>NULL</small>	<small>NULL</small>	3	33.33
	1	SIMPLE	dealer	<small>NULL</small>	eq_ref	PRIMARY, dealer_id_UNIQUE	PRIMARY	4	mydb.goods.dealer_id	1	100.00
	1	SIMPLE	office	<small>NULL</small>	eq_ref	PRIMARY, office_id_UNIQUE	PRIMARY	4	mydb.goods.office_id	1	100.00

```

EXPLAIN SELECT STRAIGHT_JOIN DISTINCT office.`name` AS office,
dealer.`name` AS dealer
FROM mydb.goods
INNER JOIN mydb.office
    ON goods.office_id = office.id
INNER JOIN mydb.dealer
    ON goods.dealer_id = dealer.id
WHERE goods.price <= 3500;

```

	id	select_type	table	partitions	type	possible_keys	key	key_len	ref	rows	filtered
►	1	SIMPLE	goods	<small>NULL</small>	ALL	fk_goods_dealer1_idx, fk_goods_offic...	<small>NULL</small>	<small>NULL</small>	<small>NULL</small>	3	33.33
	1	SIMPLE	office	<small>NULL</small>	eq_ref	PRIMARY, office_id_UNIQUE	PRIMARY	4	mydb.goods.office_id	1	100.00
	1	SIMPLE	dealer	<small>NULL</small>	eq_ref	PRIMARY, dealer_id_UNIQUE	PRIMARY	4	mydb.goods.dealer_id	1	100.00

**Висновок:** Виконуючи цю лабораторну роботу, 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.