

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ  
НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «ЛЬВІВСЬКА ПОЛІТЕХНІКА»

КАФЕДРА СИСТЕМ ШТУЧНОГО ІНТЕЛЕКТУ



**Лабораторна робота №13**

**на тему:**

**«Аналіз та оптимізація запитів»**

**з курсу:**

**«Організація баз даних та знань»**

Виконав:

ст. гр. КН-210

Холод Ігор

Львів – 2020

## Лабораторна робота №13

**Тема:** 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. Визначимо наявні індекси у таблиці drug.

```
✓ show index from drug;
```

	Table	Non-unique	Key_name	Seq_in_index	Column_name	Collation	Cardinality
1	drug	0	PRIMARY	1	id	A	8
2	drug	0	id	1	id	A	8
3	drug	0	id	2	description_id	A	8
4	drug	1	fk_Drug_Manufacturer	1	manufacturer_id	A	5
5	drug	1	fk_Drug_Description	1	description_id	A	5

Зробимо запит, який буде шукати ліки за їх назвою і дослідимо його за допомогою директиви explain:

```
✓ explain select * from drug where name = 'drug_1';
```

	id	select_type	table	partitions	type	possible_keys	key	key_len	ref	rows	filtered	Extra
1	1	SIMPLE	drug	<null>	ALL	<null>	<null>	<null>	<null>	8	12.5	Using where

2. Створимо новий індекс для поля name таблиці drug:

```
✓ create index drug_name on drug(name);
```

```
✓ show index from drug;
```

	Table	Non-unique	Key_name	Seq_in_index	Column_name	Collation	Cardinality	Sub-part	Packed	Null	Index_type
1	drug	0	PRIMARY	1	id	A	8	<null>	<null>		BTREE
2	drug	0	id	1	id	A	8	<null>	<null>		BTREE
3	drug	0	id	2	description_id	A	8	<null>	<null>		BTREE
4	drug	1	fk_Drug_Manufacturer	1	manufacturer_id	A	5	<null>	<null>		BTREE
5	drug	1	fk_Drug_Description	1	description_id	A	5	<null>	<null>		BTREE
6	drug	1	drug_name	1	name	A	8	<null>	<null>		BTREE

І повторимо запит:

	id	select_type	table	partitions	type	possible_keys	key	key_len	ref	rows	filtered	Extra
1	1	SIMPLE	drug	<null>	ref	drug_name	drug_name	482	const	1	100	<null>

Як бачимо, за допомогою індексів ми скоротили кількість рядків, які СУБД аналізує для пошуку з 8 до 1 (колонка rows)

3. Дослідимо запит з попередніх лабораторних робіт, який шукає ліки за адресою аптеки.

```
✓ explain select p.name as pharmacy_name, n.name as pharmacy_network, p.address as address, d.name as drug_name,
    m.name as drug_manufacturer, s.amount
from pharmacy p
join network n on n.id = p.network_id
join pharmacydrug s on s.pharmacy_id = p.id
join drug d on d.id = s.drug_id and d.name='drug_2'
join manufacturer m on m.id = d.manufacturer_id
where p.address='address_1';
```

id	select_type	table	partitions	type	possible_keys	key	key_len	ref
1	1 SIMPLE	p	<null>	ALL	PRIMARY,id,fk_Pharmacy_Network	<null>	<null>	<null>
2	1 SIMPLE	n	<null>	eq_ref	PRIMARY,id	PRIMARY	4	db.p.network_id
3	1 SIMPLE	s	<null>	ref	fk_PharmacyDrug_Pharmacy,fk_PharmacyDrug_Drug	fk_PharmacyDrug_Pharmacy	4	db.p.id
4	1 SIMPLE	d	<null>	eq_ref	PRIMARY,id,fk_Drug_Manufacturer	PRIMARY	4	db.s.drug_id
5	1 SIMPLE	m	<null>	eq_ref	PRIMARY,id	PRIMARY	4	db.d.manufacturer_id

  

ref	rows	filtered	Extra
<null>	4	25	Using where
db.p.network_id	1	100	<null>
db.p.id	1	100	<null>
db.s.drug_id	1	12.5	Using where
db.d.manufacturer_id	1	100	<null>

Модифікуємо наш запит, додавши індекс до поля address таблиці pharmacy, а також використавши straight\_join:

```
✓ create index pharmacy_address on pharmacy(address);
```

```
✓ explain select p.name as pharmacy_name, n.name as pharmacy_network, p.address as address, d.name as drug_name,
m.name as drug_manufacturer, s.amount
from pharmacy p
straight_join network n on n.id = p.network_id
straight_join pharmacydrug s on s.pharmacy_id = p.id
straight_join drug d on d.id = s.drug_id and d.name='drug_2'
straight_join manufacturer m on m.id = d.manufacturer_id
where p.address='address_1';
```

І повторимо цей запит:

id	select_type	table	partitions	type	possible_keys	key	key_len	ref
1	1 SIMPLE	p	<null>	ref	PRIMARY,id,fk_Pharmacy_Network,pharmacy_address	pharmacy_address	402	const
2	1 SIMPLE	n	<null>	eq_ref	PRIMARY,id	PRIMARY	4	db.p.network_id
3	1 SIMPLE	s	<null>	ref	fk_PharmacyDrug_Pharmacy,fk_PharmacyDrug_Drug	fk_PharmacyDrug_Pharmacy	4	db.p.id
4	1 SIMPLE	d	<null>	eq_ref	PRIMARY,id,fk_Drug_Manufacturer,drug_name	PRIMARY	4	db.s.drug_id
5	1 SIMPLE	m	<null>	eq_ref	PRIMARY,id	PRIMARY	4	db.d.manufacturer_id

  

ref	rows	filtered	Extra
const	1	100	<null>
db.p.network_id	1	100	<null>
db.p.id	1	100	<null>
db.s.drug_id	1	12.5	Using where
db.d.manufacturer_id	1	100	<null>

**Висновок:** я навчився досліджувати та оптимізовувати запити у СУБД MySQL з використанням індексів та straight\_join.