

Міністерство освіти і науки України
Національний університет «Львівська політехніка»
Інститут комп'ютерних наук та інформаційних технологій
Кафедра «Системи штучного інтелекту»

Лабораторна робота №13
З предмету: «Організація баз даних та знань»

Виконав студент

групи КН-208

Бердник Д.І.

Прийняла

Мельникова Н.І.

Львів 2020

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 пришвидшення.

Хід роботи

1. Визначити індекси таблиці.
2. Створити додаткові індекси для таблиці.
3. Дослідити процес виконання запитів за допомогою EXPLAIN

1. Визначимо індекси таблиць pharmacy, medicament та session за допомогою команди SHOW INDEX FROM <TABLE NAME>

```
show index from medicament;  
show index from pharmacy;
```

Table	Non_unique	Key_name	Seq_in_index	Column_name	Collation	Cardinality	Sub_part	Packed	Null	Index_type	Comment	Index_comment	Visible	Expression
pharmacy	0	PRIMARY	1	id_pharmacy	A	5	NULL	NULL		BTREE			YES	NULL

Table	Non_unique	Key_name	Seq_in_index	Column_name	Collation	Cardinality	Sub_part	Packed	Null	Index_type	Comment	Index_comment	Visible	Expression
medicament	0	PRIMARY	1	id_medicament	A	3	NULL	NULL		BTREE			YES	NULL

2. Створимо нові індекси для таблиць session, medicament, pharmacy. У БД є декілька запитів, які здійснюють вибірку даних за ім'ям медикаменту (medicament.name) і назвою аптеки (pharmacy.name). Створення індексів для цих полів повинно оптимізувати виконання запитів.

```
create index pharm_idx on pharmacy (id_pharmacy, name);  
create index med_idx on medicament (id_medicament, name);
```

	Table	Non_unique	Key_name	Seq_in_index	Column_name	Collation	Cardinality	Sub_part	Packed	Null	Index_type	Comment	Index_comment	Visible	Expression
►	pharmacy	0	PRIMARY	1	id_pharmacy	A	5	NULL	NULL		BTREE			YES	NULL
	pharmacy	1	pharm_idx	1	id_pharmacy	A	5	NULL	NULL		BTREE			YES	NULL
	pharmacy	1	pharm_idx	2	name	A	5	NULL	NULL		BTREE			YES	NULL

	Table	Non_unique	Key_name	Seq_in_index	Column_name	Collation	Cardinality	Sub_part	Packed	Null	Index_type	Comment	Index_comment	Visible	Expression
►	medicament	0	PRIMARY	1	id_medicament	A	3	NULL	NULL		BTREE			YES	NULL
	medicament	1	med_idx	1	id_medicament	A	10	NULL	NULL		BTREE			YES	NULL
	medicament	1	med_idx	2	name	A	10	NULL	NULL		BTREE			YES	NULL

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

```

explain select `order`.total_price, medicament.name as medicament_name, medicament.active_substance,
pharmacy.name as pharmacy_name, `session`.last_seen as date_order
from Specific_pharmacy
inner join `order`
on `order`.id_specific_pharmacy = Specific_pharmacy.id_specific_pharmacy
inner join pharmacy
on pharmacy.id_pharmacy = Specific_pharmacy.id_pharmacy
inner join medicament
on medicament.id_medicament = Specific_pharmacy.id_medicament
inner join `session`
on `session`.id_session = `order`.id_session;

```

Result Grid													Filter Rows:	Export:	Wrap Cell Content:
	id	select_type	table	partitions	type	possible_keys	key	key_len	ref	rows	filtered	Extra			
▶	1	SIMPLE	session	NULL	ALL	PRIMARY	NULL	NULL	NULL	5	100.00	NULL			
	1	SIMPLE	order	NULL	ALL	Order_fk0,Order_fk1	NULL	NULL	NULL	6	33.33	Using where; Using join buffer (Block Ne...			
	1	SIMPLE	Specific_pharmacy	NULL	eq_ref	PRIMARY,Specific_Pharmacy_fk0,Specific_Phar...	PRIMARY	4	pharmacy2.order_id_specific_pharmacy	1	100.00	NULL			
	1	SIMPLE	pharmacy	NULL	eq_ref	PRIMARY,pharm_idx	PRIMARY	4	pharmacy2.Specific_pharmacy_id_pharmacy	1	100.00	NULL			
	1	SIMPLE	medicament	NULL	eq_ref	PRIMARY,med_idx	PRIMARY	4	pharmacy2.Specific_pharmacy_id_medicament	1	100.00	NULL			

```

explain select straight_join `order`.total_price, medicament.name as medicament_name, medicament.active_substance,
pharmacy.name as pharmacy_name, `session`.last_seen as date_order
from Specific_pharmacy
inner join `order`
on `order`.id_specific_pharmacy = Specific_pharmacy.id_specific_pharmacy
inner join pharmacy
on pharmacy.id_pharmacy = Specific_pharmacy.id_pharmacy
inner join medicament
on medicament.id_medicament = Specific_pharmacy.id_medicament
inner join `session`
on `session`.id_session = `order`.id_session;

```

Result Grid										
Filter Rows:										
Export: Wrap Cell Content: T										
	id	select_type	table	partitions	type	possible_keys	key	key_len	ref	Extra
►	1	SIMPLE	Specific_pharmacy	NULL	ALL	PRIMARY,Specific_Pharmacy_fk0,Specific_Phar...	NULL	NULL	NULL	15 100.00 NULL
	1	SIMPLE	order	NULL	ref	Order_fk0,Order_fk1	Order_fk0	4	pharmacy2.Specific_pharmacy.id_specific_phar...	1 100.00 Using where
	1	SIMPLE	pharmacy	NULL	eq_ref	PRIMARY_pharm_idx	PRIMARY	4	pharmacy2.Specific_pharmacy.id_pharmacy	1 100.00 NULL
	1	SIMPLE	medicament	NULL	eq_ref	PRIMARY_med_idx	PRIMARY	4	pharmacy2.Specific_pharmacy.id_medicament	1 100.00 NULL
	1	SIMPLE	session	NULL	eq_ref	PRIMARY	PRIMARY	4	pharmacy2.order.id_session	1 100.00 NULL

Висновок

На цій лабораторній роботі я навчився аналізувати роботу СУБД та оптимізовувати виконання складних запитів на вибірку даних