

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

Лабораторна робота №13

з дисципліни
«Організація баз даних та знань»

Виконав:
студент групи КН-207
Белан В.Ю

Прийняла:
Мельникова Н.І.

Львів – 2019 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. За допомогою директиви SHOW INDEX визначимо наявні індекси для таблиць Train, Schedule і Stations.

```
1 • use metro;
2
3 • show index from train;
4 • show index from schedule;
5 • show index from station;
```

Table	Non_unique	Key_name	Seq_in_index	Column_name	Collation	Cardinality	Sub_part	Packed	Null	Index_type	Comment	Index_comment	Visible	Expression
train	0	PRIMARY	1	id	A	30	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
schedule	0	PRIMARY	1	id	A	60	NULL	NULL		BTREE			YES	NULL
schedule	0	PRIMARY	2	Station_id	A	60	NULL	NULL		BTREE			YES	NULL
schedule	0	PRIMARY	3	Train_id	A	60	NULL	NULL		BTREE			YES	NULL
schedule	0	PRIMARY	4	Line_id	A	60	NULL	NULL		BTREE			YES	NULL
schedule	1	fk_Schedule_Station	1	Station_id	A	12	NULL	NULL		BTREE			YES	NULL
schedule	1	fk_Schedule_Train1	1	Train_id	A	15	NULL	NULL		BTREE			YES	NULL
schedule	1	fk_Schedule_Line1	1	Line_id	A	3	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
station	0	PRIMARY	1	id	A	30	NULL	NULL		BTREE			YES	NULL

2. Створимо нові індекси для таблиць Train, Station і Schedule.

```
1 • use metro;
2
3 • create index stationINDX3 on station (id, name);
4
```

```
1 • use metro;
2
3 • create index trainINDX on train (id, capacity, passenger);
4
```

```

1 • use metro;
2
3 • create index scheduleINDX on schedule (train_id, station_id);
4

```

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

```

1 • use metro;
2
3 • explain select station.name, schedule.timer
4     from station inner join schedule
5     on station.id = schedule.station_id
6     where schedule.train_id in (select train.id from train where train.passenger >=1200)
7     and station.name = 'Petrivka';

```

	id	select_type	table	partitions	type	possible_keys	key	key_len	ref	rows	filtered	Extra
►	1	SIMPLE	station	NULL	index	PRIMARY,stationINDX3	stationINDX3	186	NULL	30	10.00	Using where; Using index
	1	SIMPLE	schedule	NULL	ref	fk_Schedule_Station,scheduleINDX	fk_Schedule_Station	4	metro.station.id	5	100.00	Using where; Using index
	1	SIMPLE	train	NULL	eq_ref	PRIMARY,trainINDX	PRIMARY	4	metro.schedule.Train_id	1	33.33	Using where

```

1 • use metro;
2
3 • explain select straight_join station.name, schedule.timer
4     from station inner join schedule
5     on station.id = schedule.station_id
6     where schedule.train_id in (select train.id from train where train.passenger >=1200)
7     and station.name = 'Petrivka';

```

	id	select_type	table	partitions	type	possible_keys	key	key_len	ref	rows	filtered	Extra
►	1	PRIMARY	station	NULL	index	PRIMARY,stationINDX3	stationINDX3	186	NULL	30	10.00	Using where; Using index
	1	PRIMARY	schedule	NULL	ref	fk_Schedule_Station	fk_Schedule_Station	4	metro.station.id	5	100.00	Using where
	2	DEPENDENT SUBQUERY	train	NULL	unique_subquery	PRIMARY,trainINDX	PRIMARY	4	func	1	33.33	Using where

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.