

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

Кафедра систем штучного інтелекту

Лабораторна робота №13

з дисципліни
«Організація баз даних та знань»

Виконав:

студент групи КН-208

Гірник Ю.В.

Викладач:

к.т.н.

Мельникова Н.І.

Львів – 2020 р.

Лабораторна робота №13

Аналіз та оптимізація запитів

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

1. За допомогою директиви **SHOW INDEX** визначимо наявні індекси для таблиць Tour і Agency.

SHOW INDEX FROM agency;

	Table	Non_unique	Key_name	Seq_in_index	Column_name	Collation	Cardinality	Sub_part	Packed	Null	Index_type	Comment	Index_comment	Visible	Expression
►	agency	0	PRIMARY	1	agencyId	A	4	NULL	NULL		BTREE			YES	NULL
	agency	0	phone	1	phone	A	4	NULL	NULL		BTREE			YES	NULL
	agency	0	web_site	1	web_site	A	4	NULL	NULL		BTREE			YES	NULL

SHOW INDEX FROM tour;

	Table	Non_unique	Key_name	Seq_in_index	Column_name	Collation	Cardinality	Sub_part	Packed	Null	Index_type	Comment	Index_comment	Visible	Expression
►	tour	0	PRIMARY	1	tourId	A	6	NULL	NULL		BTREE			YES	NULL
	tour	1	tour_agency_fk	1	agency_id	A	3	NULL	NULL		BTREE			YES	NULL

Виконаємо аналіз виконання складного запиту використовуючи **EXPLAIN** та опцію **STRAIGHT_JOIN**.

```
EXPLAIN SELECT aname AS 'Agence name', TNAME AS 'Tour  
name', COUNT(DISTINCT(client_tour.client_id)) AS  
'Clients'  
FROM agency INNER JOIN tour  
INNER JOIN client_tour  
    ON agency.agencyId = tour.agency_id  
    AND tour.tourId = client_tour.tour_id  
WHERE aname = 'HachuNaBali' AND tour.start_date BETWEEN  
    '2020-07-10' AND '2020-08-26'  
GROUP BY tname
```

id	select_type	table	partitions	type	possible_keys	key	key_len	ref	rows	filtered	Extra
1	SIMPLE	agency	NULL	ALL	PRIMARY	NULL	NULL	NULL	4	25.00	Using where; Using temp
1	SIMPLE	tour	NULL	ALL	PRIMARY, tour_agency_fk	NULL	NULL	NULL	6	16.67	Using where; Using join b
1	SIMPLE	client_tour	NULL	ref	client_tour_tours_fk	client_tour_tours_fk	4	travel_agency.tour.tourId	4	100.00	NULL

```

EXPLAIN SELECT STRAIGHT_JOIN aname AS 'Agence name',
      TNAME AS 'Tour name',
      COUNT(DISTINCT(client_tour.client_id)) AS 'Clients'
FROM agency INNER JOIN tour
INNER JOIN client_tour
      ON agency.agencyId = tour.agency_id
      AND tour.tourId = client_tour.tour_id
WHERE aname = 'HachuNaBali' AND tour.start_date BETWEEN
      '2020-07-10' AND '2020-08-26'
GROUP BY tname

```

id	select_type	table	partitions	type	possible_keys	key	key_len	ref	rows	filtered	Extra
1	SIMPLE	agency	NULL	ALL	PRIMARY	NULL	NULL	NULL	4	25.00	Using where; Using tempor
1	SIMPLE	tour	NULL	ALL	PRIMARY,tour_agency_fk	NULL	NULL	NULL	6	16.67	Using where; Using join bu
1	SIMPLE	client_tour	NULL	ref	client_tour_tours_fk	client_tour_tours_fk	4	travel_agency.tour.tourId	4	100.00	NULL

2. Створимо новий індекс для таблиці Tour і Agency. Часто можливі запити, які здійснюють вибірку даних за назвою агенції(поле aname), за датою початку туру (поле start_date). Створення індексів для цих полів повинно оптимізувати виконання запитів.

```

CREATE INDEX anameIDX ON agency (agencyId, aname);
SHOW INDEX FROM agency;

```

Table	Non_unique	Key_name	Seq_in_index	Column_name	Collation	Cardinality	Sub_part	Packed	Null	Index_type	Comment	Index_comme	Visible	Expression
agency	0	PRIMARY	1	agencyId	A	4	NULL	NULL		BTREE			YES	NULL
agency	0	phone	1	phone	A	4	NULL	NULL		BTREE			YES	NULL
agency	0	web_site	1	web_site	A	4	NULL	NULL		BTREE			YES	NULL
agency	1	anameIDX	1	agencyId	A	3	NULL	NULL		BTREE			YES	NULL
agency	1	anameIDX	2	aname	A	3	NULL	NULL		BTREE			YES	NULL

```

CREATE INDEX tdateIDX ON tour (tourId, start_date);
SHOW INDEX FROM tour;

```

Table	Non_unique	Key_name	Seq_in_index	Column_name	Collation	Cardinality	Sub_part	Packed	Null	Index_type	Comment	Index_comme	Visible	Expression
tour	0	PRIMARY	1	tourId	A	6	NULL	NULL		BTREE			YES	NULL
tour	1	tour_agency_fk	1	agency_id	A	3	NULL	NULL		BTREE			YES	NULL
tour	1	tdateIDX	1	tourId	A	6	NULL	NULL		BTREE			YES	NULL
tour	1	tdateIDX	2	start_date	A	6	NULL	NULL		BTREE			YES	NULL

3. Виконаємо аналіз виконання складного запиту використовуючи **EXPLAIN** та опцію **STRAIGHT_JOIN**.

```

EXPLAIN SELECT aname AS 'Agence name', TNAME AS 'Tour
      name', COUNT(DISTINCT(client_tour.client_id)) AS
      'Clients'
FROM agency INNER JOIN tour
INNER JOIN client_tour
      ON agency.agencyId = tour.agency_id
      AND tour.tourId = client_tour.tour_id
WHERE aname = 'HachuNaBali' AND tour.start_date BETWEEN
      '2020-07-10' AND '2020-08-26'
GROUP BY tname

```

id	select_type	table	partitions	type	possible_keys	key	key_len	ref	rows	filtered	Extra
1	SIMPLE	tour	NULL	ALL	PRIMARY,tour_agency_fk,tdateIDX	NULL	NULL	NULL	6	16.67	Using where; Using filesort
1	SIMPLE	agency	NULL	eq_ref	PRIMARY,anameIDX	PRIMARY	4	travel_agency.tour.agency_id	1	25.00	Using where
1	SIMPLE	client_tour	NULL	ref	client_tour_tours_fk	client_tour_tours_fk	4	travel_agency.tour.tourId	4	100.00	NULL

```

EXPLAIN SELECT STRAIGHT_JOIN aname AS 'Agence name',
      TNAME AS 'Tour name',
      COUNT(DISTINCT(client_tour.client_id)) AS 'Clients'
FROM agency INNER JOIN tour
INNER JOIN client_tour
      ON agency.agencyId = tour.agency_id
      AND tour.tourId = client_tour.tour_id
WHERE aname = 'HachuNaBali' AND tour.start_date BETWEEN
      '2020-07-10' AND '2020-08-26'
GROUP BY tname

```

id	select_type	table	partitions	type	possible_keys	key	key_len	ref	rows	filtered	Extra
1	SIMPLE	agency	NULL	index	PRIMARY,anameIDX	anameIDX	86	NULL	4	25.00	Using where; Using index;
1	SIMPLE	tour	NULL	ref	PRIMARY,tour_agency_fk,tdateIDX	tour_agency_fk	4	travel_agency.agency.agenc...	2	16.67	Using where
1	SIMPLE	client_tour	NULL	ref	client_tour_tours_fk	client_tour_tours_fk	4	travel_agency.tour.tourId	4	100.00	NULL

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.