

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



Звіт
до лабораторної роботи № 13
з дисципліни «ОБДЗ»

Виконала:
студентка групи КН-211
Піпенко Анастасія
Прийняла:
асистент
Якимишин Х.М.

Львів-2020

Тема: Аналіз та оптимізація запитів.

Мета: 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:

- За допомогою директиви SHOW INDEX визначаю наявні індекси для таблиць student і teacher.

```
1 • SHOW INDEX FROM student;
```

Result Grid													
Table	Non_unique	Key_name	Seq_in_index	Column_name	Collation	Cardinality	Sub_part	Packed	Null	Index_type	Comment	Index_comment	Visible
student	0	PRIMARY	1	student_id	A	10	NULL	NULL	NULL	BTREE			YES
student	1	idx_student_name	1	last_name	A	10	NULL	NULL	NULL	BTREE			YES

```
1 • SHOW INDEX FROM teacher;
```

Result Grid													
Table	Non_unique	Key_name	Seq_in_index	Column_name	Collation	Cardinality	Sub_part	Packed	Null	Index_type	Comment	Index_comment	Visible
teacher	0	PRIMARY	1	teacher_id	A	3	NULL	NULL	NULL	BTREE			YES

- Виконуємо аналіз виконання запиту, використовуючи EXPLAIN:

```
explain SELECT student.last_name, groupx.group_id, teacher.last_name,
course.c_language
FROM (((student INNER JOIN group_students) INNER JOIN groupx) INNER JOIN course) INNER
JOIN teacher
ON student.student_id = group_students.student_id
AND group_students.group_id = groupx.group_id
AND groupx.course_id = course.course_id
AND groupx.teacher_id = teacher.teacher_id
WHERE student.last_name = 'Aber';
```

	id	select_type	table	partitions	type	possible_keys	key	key_len	ref	rows	filtered	Extra
▶	1	SIMPLE	student	HULL	ALL	PRIMARY	HULL	HULL	HULL	10	10.00	Using temporary; Using filesort
1	SIMPLE	group_students		HULL	ref	group_students_fk0,group_students_fk1	group_students_fk1	4	language_school.student.student_id	1	100.00	Using where
1	SIMPLE	groupx		HULL	eq_ref	PRIMARY,group_fk0,group_fk1	PRIMARY	4	language_school.group_students.group_id	1	100.00	Using where
1	SIMPLE	teacher		HULL	eq_ref	PRIMARY	PRIMARY	4	language_school.groupx.teacher_id	1	100.00	Using where
1	SIMPLE	course		HULL	eq_ref	PRIMARY	PRIMARY	4	language_school.groupx.course_id	1	100.00	Using where

3. Оскільки ми маємо тип ALL, потрібно оптимізувати запит. Для цього створюю новий індекс.

```
CREATE INDEX idx_student_name ON student (last_name);
```

	id	select_type	table	partitions	type	possible_keys	key	key_len	ref	rows	filtered
▶	1	SIMPLE	student	HULL	ref	PRIMARY,idx_student_name	idx_student_name	1022	const	1	100.00
1	SIMPLE	group_students		HULL	ref	group_students_fk0,group_students_fk1	group_students_fk1	4	language_school.student.student_id	1	100.00
1	SIMPLE	groupx		HULL	eq_ref	PRIMARY,group_fk0,group_fk1	PRIMARY	4	language_school.group_students.group_id	1	100.00
1	SIMPLE	teacher		HULL	eq_ref	PRIMARY	PRIMARY	4	language_school.groupx.teacher_id	1	100.00
1	SIMPLE	course		HULL	eq_ref	PRIMARY	PRIMARY	4	language_school.groupx.course_id	1	100.00

Висновок: на цій лабораторній роботі я навчилась аналізувати і оптимізувати виконання запитів. Для аналізу запитів було використано EXPLAIN, а для оптимізації – модифікація порядку з'єднання (straight_join) таблиць і створення додаткових індексів.