

МІНІСТЕРСТВО ОСВІТИ І НАУКИ УКРАЇНИ НАЦІОНАЛЬНИЙ
УНІВЕРСИТЕТ «ЛЬВІВСЬКА ПОЛІТЕХНІКА»

Інститут комп'ютерних наук та інформаційних технологій

Кафедра систем штучного інтелекту



Лабораторна робота №13

на тему “Аналіз та оптимізація запитів”

Виконав:

Романишин М.Р.

КН-211

Викладач:

Якимишин Х.М.

Львів – 2020

Лабораторна робота №13
з курсу “ОБДЗ”
на тему:
“Аналіз та оптимізація запитів”

Мета роботи: Розробити SQL запити, які моделюють роботу тригерів: каскадне знищення, зміна та доповнення записів у зв’язаних таблицях.

Хід роботи

1. Виконаємо аналіз складного запиту з минулої лабораторної роботи.

```
explain SELECT straight_join user.login, user_project_task.user_role,  
task.description, task.title, task.deadline_time  
FROM user inner JOIN user_project_task  
INNER JOIN task  
ON user.user_id = user_project_task.user_id  
AND task.task_id = user_project_task.task_id  
WHERE task.deadline_time between '2018-01-01 10:34:09' and '2018-12-30 10:34:09'  
and length(user.login) > 5;
```

Result Grid												
Filter Rows: <input type="text"/>												
Exports: <input type="button" value="Excel"/> Wrap Cell Content: <input type="button" value="15"/>												
	id	select_type	table	partitions	type	possible_keys	key	key_len	ref	rows	filtered	Extra
▶	1	SIMPLE	task	NULL	ALL	PRIMARY	NULL	NULL	NULL	15	11.11	Using where
	1	SIMPLE	user_project_task	NULL	ref	user_id,task_id	task_id	5	projectmanagementsystem.task.task_id	2	100.00	Using where
	1	SIMPLE	user	NULL	eq_ref	PRIMARY	PRIMARY	4	projectmanagementsystem.user_project_task....	1	100.00	Using where

2. Створимо індекси для таблиць User та Task. В базі є багато запитів які використовують поля login та deadline_time. Створення індексів для них повинно оптимізувати запити.

```
create index task_indx on task(deadline_time);  
create unique index user_indx on user(login);  
show index from task;  
show index from user;
```

Result Grid														
Filter Rows:														
Exports: Wrap Cell Content: 15														
	Table	Non_unique	Key_name	Seq_in_index	Column_name	Collation	Cardinality	Sub_part	Packed	Null	Index_type	Comment	Index_comment	Visible
▶	task	0	PRIMARY	1	task_id	A	0	NULL	NULL	NULL	BTREE			YES
	task	1	task_indx	1	deadline_time	A	14	NULL	NULL	NULL	BTREE			YES

Result Grid														
Filter Rows:														
Exports: Wrap Cell Content: 15														
	Table	Non_unique	Key_name	Seq_in_index	Column_name	Collation	Cardinality	Sub_part	Packed	Null	Index_type	Comment	Index_comment	Visible
▶	user	0	PRIMARY	1	user_id	A	1	NULL	NULL	NULL	BTREE			YES
	user	0	login	1	login	A	1	NULL	NULL	NULL	BTREE			YES
	user	0	user_indx	1	login	A	8	NULL	NULL	NULL	BTREE			YES
	user	1	user_contacts	1	contacts_id	A	1	NULL	NULL	YES	BTREE			YES

3. Запит після створення індексів.

Result Grid												
Filter Rows:												
Export: Wrap Cell Content: 15												
	id	select_type	table	partitions	type	possible_keys	key	key_len	ref	rows	filtered	Extra
▶	1	SIMPLE	user	NULL	index	PRIMARY	login	162	NULL	8	100.00	Using index
	1	SIMPLE	user_project_task	NULL	ref	user_id,task_id	user_id	5	projectmanagementsystem.user.user_id	2	100.00	Using where
	1	SIMPLE	task	NULL	eq_ref	PRIMARY,task_indx	PRIMARY	4	projectmanagementsystem.user_project_task.t...	1	46.67	Using where

Висновок: Розробити SQL запити, які моделюють роботу тригерів: каскадне знищення, зміна та доповнення записів у зв’язаних таблицях.