**МІНІСТЕРСТВО ОСВІТИ І НАУКИ**

**НАЦІОНАЛЬНИЙ УНІВЕРСИТЕТ «ЛЬВІВСЬКА ПОЛІТЕХНІКА»**

**Інститут комп'ютерних наук та інформаційних технологій**

**Кафедра систем штучного інтелекту**

****

**Звіт до лабораторної роботи №13**

з дисципліни

“Організація баз даних та знань”

**Виконав**:

ст. гр. КН-211

Петров Дмитро

**Викладач:**

Якимишин Х.М.

Львів – 2020

**Тема:** 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. Проаналізувати ефективність запитів.
3. Створити додаткові індекси для таблиці.
4. Дослідити процес виконання запитів за допомогою EXPLAIN.
5. Визначити індекси таблиці.

SHOW INDEX FROM user;



1. Проаналізувати ефективність запитів.

EXPLAIN SELECT group\_name AS category, COUNT(db\_sht.order.id) AS amount

FROM (((db\_sht.group INNER JOIN article\_group)

INNER JOIN goods)

INNER JOIN db\_sht.order)

INNER JOIN db\_sht.user ON db\_sht.user.email = '1233@hots.com'

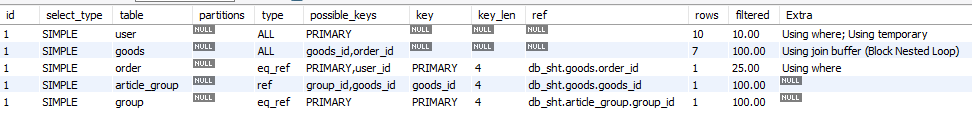
AND user.id = db\_sht.order.user\_id

AND db\_sht.order.id = goods.order\_id

AND goods.goods\_id = article\_group.goods\_id

AND article\_group.group\_id = db\_sht.group.id

WHERE TO\_DAYS(CURRENT\_DATE) - TO\_DAYS(db\_sht.order.ordered) <= 31

GROUP BY category;   


EXPLAIN SELECT group\_name AS category, COUNT(db\_sht.order.id) AS amount

FROM (((db\_sht.group INNER JOIN article\_group)

INNER JOIN goods)

INNER JOIN db\_sht.order)

INNER JOIN db\_sht.user ON db\_sht.user.email = '1233@hots.com'

AND user.id = db\_sht.order.user\_id

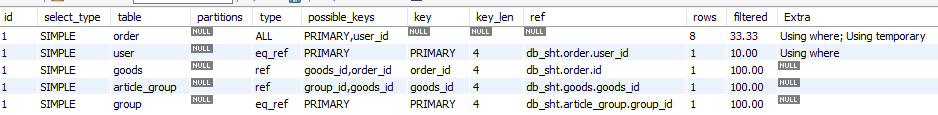
AND db\_sht.order.id = goods.order\_id

AND goods.goods\_id = article\_group.goods\_id

AND article\_group.group\_id = db\_sht.group.id

WHERE db\_sht.order.ordered >= DATE\_SUB(CURRENT\_DATE, INTERVAL 31 DAY)

GROUP BY category;



Як бачимо в залежності від формування функції вибору запиту, сама вибірка може проводитися. Про це свідчить поле type, що відображає по яких полях відбувається вибірка. Втаблиці ордер вона має проводитися по одному полю. Також це видно в полі Extras, де вказані дії, що відбувається з кожною таблицею при вибірці – використовується тимчасове з’єднання, що сповільнює операцію вибірки.

1. Створити додаткові індекси для таблиці.

SHOW INDEX FROM user;

ALTER TABLE user MODIFY email varchar(50);

CREATE UNIQUE INDEX user\_coupled\_index\_1 ON user (id, email);

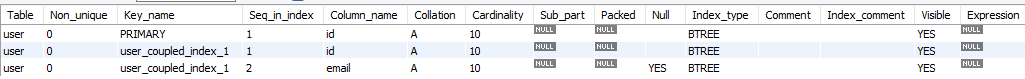
-- ALTER TABLE order MODIFY email varchar(50);

CREATE UNIQUE INDEX order\_index\_1 ON db\_sht.order (id);

alter table goods drop index goods\_index\_1;

CREATE UNIQUE INDEX goods\_index\_1 ON goods (id, order\_id);

SHOW INDEX FROM goods;



1. Дослідити процес виконання запитів за допомогою EXPLAIN.

EXPLAIN SELECT group\_name AS category, COUNT(db\_sht.order.id) AS amount

FROM (((db\_sht.group INNER JOIN article\_group)

INNER JOIN goods)

INNER JOIN db\_sht.order)

INNER JOIN db\_sht.user ON db\_sht.user.email = '1233@hots.com'

AND user.id = db\_sht.order.user\_id

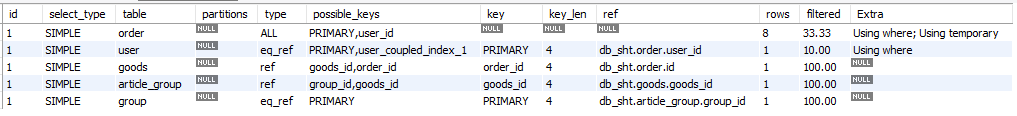
AND db\_sht.order.id = goods.order\_id

AND goods.goods\_id = article\_group.goods\_id

AND article\_group.group\_id = db\_sht.group.id

WHERE TO\_DAYS(CURRENT\_DATE) - TO\_DAYS(db\_sht.order.ordered) <= 31

GROUP BY category;





EXPLAIN SELECT group\_name AS category, COUNT(db\_sht.order.id) AS amount

FROM (((db\_sht.group INNER JOIN article\_group)

INNER JOIN goods)

INNER JOIN db\_sht.order)

INNER JOIN db\_sht.user ON db\_sht.user.email = '1233@hots.com'

AND user.id = db\_sht.order.user\_id

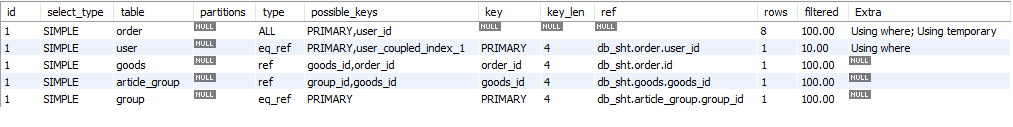
AND db\_sht.order.id = goods.order\_id

AND goods.goods\_id = article\_group.goods\_id

AND article\_group.group\_id = db\_sht.group.id

WHERE db\_sht.order.ordered >= DATE\_SUB(CURRENT\_DATE, INTERVAL 31 DAY)

GROUP BY category;



В результаті в обох варіантах вибірки ми отримали оптимальний пошук.

**Висновок** : На даній лабораторній роботі я навчився аналізувати і оптимізувати виконання запитів. Для аналізу запитів було використано директиву EXPLAIN, а для оптимізації – створення додаткових індексів.