

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:**

Свистович С.А.

КН-211

**Викладач:**

Якимишин Х.М.

Львів – 2020

Лабораторна робота №13  
з курсу “ОБДЗ”  
на тему:  
**“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. Створення індексів:

```
create index artist_index on artist(idArtist,BirhDate);
create index song_index on song(artist_id,idSong,Name);
create index on_song_platform_index on
song_platform(song_id,platform_id);
create index on_platform_index on platform(idPlatform);
```

2. Приклад straight joiny:

```
explain select straight_join
artist.Nickname,artist.BirhDate, song.Name
FROM artist inner join song inner join song_platform
inner join platform
on artist.idArtist = song.artist_id and
song.idSong=song_platform.song_id and
song_platform.platform_id = platform.idPlatform and
length(song.Name)>2;
```

3. Приклад explainy joiny:

```
explain select artist.Nickname,artist.BirhDate, song.Name
FROM artist inner join song inner join song_platform
inner join platform
on artist.idArtist = song.artist_id and
song.idSong=song_platform.song_id and
song_platform.platform_id = platform.idPlatform and
length(song.Name)>2;
```

id	select_type	table	partitions	type	possible_keys	key	key_len	ref	rows	filtered	Extra
1	SIMPLE	platform	NULL	index	PRIMARY, on_platform_index	on_platform_index	4	NULL	4	100.00	Using index
1	SIMPLE	song_platform	NULL	ref	fk_song_platform_song1_idx, fk_song_platform...	fk_song_platform_platform1_idx	4	rating_of_musicians.platform.idPlatform	1	100.00	Using index
1	SIMPLE	song	NULL	eq_ref	PRIMARY, fk_song_artist1_idx, song_index	PRIMARY	4	rating_of_musicians.song_platform.song_id	1	100.00	Using where
1	SIMPLE	artist	NULL	eq_ref	PRIMARY, idArtist_UNIQUE, artist_index	PRIMARY	4	rating_of_musicians.song.artist_id	1	100.00	Using index

## SQL-скріпт

```
explain select artist.Nickname,artist.BirhDate, song.Name
FROM artist inner join song inner join song_platform inner join platform
on artist.idArtist = song.artist_id and song.idSong=song_platform.song_id and
song_platform.platform_id = platform.idPlatform;
```

```
drop index artist_index on artist;
drop index song_index on song;
drop index on_song_platform_index on song_platform;
drop index on_platform_index on platform;
```

```
create index artist_index on artist(idArtist,BirhDate);
create index song_index on song(artist_id,idSong,Name);
create index on_song_platform_index on song_platform(song_id,platform_id);
create index on_platform_index on platform(idPlatform);
```

```
show index from artist;
show index from song;
```

```
explain select straight_join artist.Nickname,artist.BirhDate, song.Name
FROM artist inner join song inner join song_platform inner join platform
on artist.idArtist = song.artist_id and song.idSong=song_platform.song_id and
song_platform.platform_id = platform.idPlatform and length(song.Name)>2;
```

```
explain select artist.Nickname,artist.BirhDate, song.Name
FROM artist inner join song inner join song_platform inner join platform
on artist.idArtist = song.artist_id and song.idSong=song_platform.song_id and
song_platform.platform_id = platform.idPlatform and length(song.Name)>2;
```

**Висновок:** під час виконання даної лабораторної роботи, я навчився 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.