

MODUL PRAKTIKUM BASIS DATA

INDEXING



ITERA

**Teknik Informatika
Institut Teknologi Sumatera
Tahun 2020**

MODUL 1

INDEXING



ITERA

Oleh

Dosen Pengampu :

Rajif Agung Yunmar, S.Kom., M.Cs.

Hartanto Tantriawan, S.Kom., M.Kom.

Asisten Praktikum

Rivaldo Fernandes (14117103)

Bagas Pangestu (14117026)

Maria Andini (14117041)

Afdi Fauzul Bahar (14117149)

Agus Michael Sianipar (14117066)

Nicolaus Edwardo Felix (14117127)

Muhammad Nur Faqqih (14117168)

Muhammad Muttaqin (14117138)



INDEXING

A. Indexing Database

Pernahkah kalian mendengar istilah index? Biasanya istilah ini ada didalam buku. Fungsinya yaitu memudahkan kita mencari suatu kata yang ada didalam buku tersebut dengan mencantumkan nomor halaman dimana kata itu berada. Misal, kita ingin mencari suatu kata 'lorem' , namun didalam buku tersebut tidak ada index, maka yang akan dilakukan adalah membaca dari halaman awal sampai kata 'lorem' ditemukan (ini akan membutuhkan banyak waktu jika kata 'lorem' ada di halaman akhir). Namun, jika pada buku tersebut terdapat index, maka kita hanya perlu membuka nomor halamannya saja untuk menemukan kata 'lorem'.

Begitu pula pada database, index berguna sebagai penanda suatu data ditempatkan, sehingga pada saat melakukan pencarian, waktu proses yang dibutuhkan akan lebih cepat. Lalu, apa itu indexing pada database? Indexing pada database yaitu proses membuat index.

Index merupakan objek struktur data tersendiri yang tidak bergantung kepada struktur tabel. Setiap index terdiri dari nilai kolom dan penunjuk (atau ROWID) ke baris yang berisi nilai tersebut. Penunjuk tersebut secara langsung menunjuk ke baris yang tepat pada tabel, sehingga menghindari terjadinya full table-scan. Akan tetapi lebih banyak index pada tabel tidak berarti akan mempercepat query. Semakin banyak index pada suatu tabel menyebabkan kelambatan pemrosesan perintah-perintah DML (Data Manipulation Language), karena setiap terjadi perubahan data maka index juga harus disesuaikan.



B. Latihan

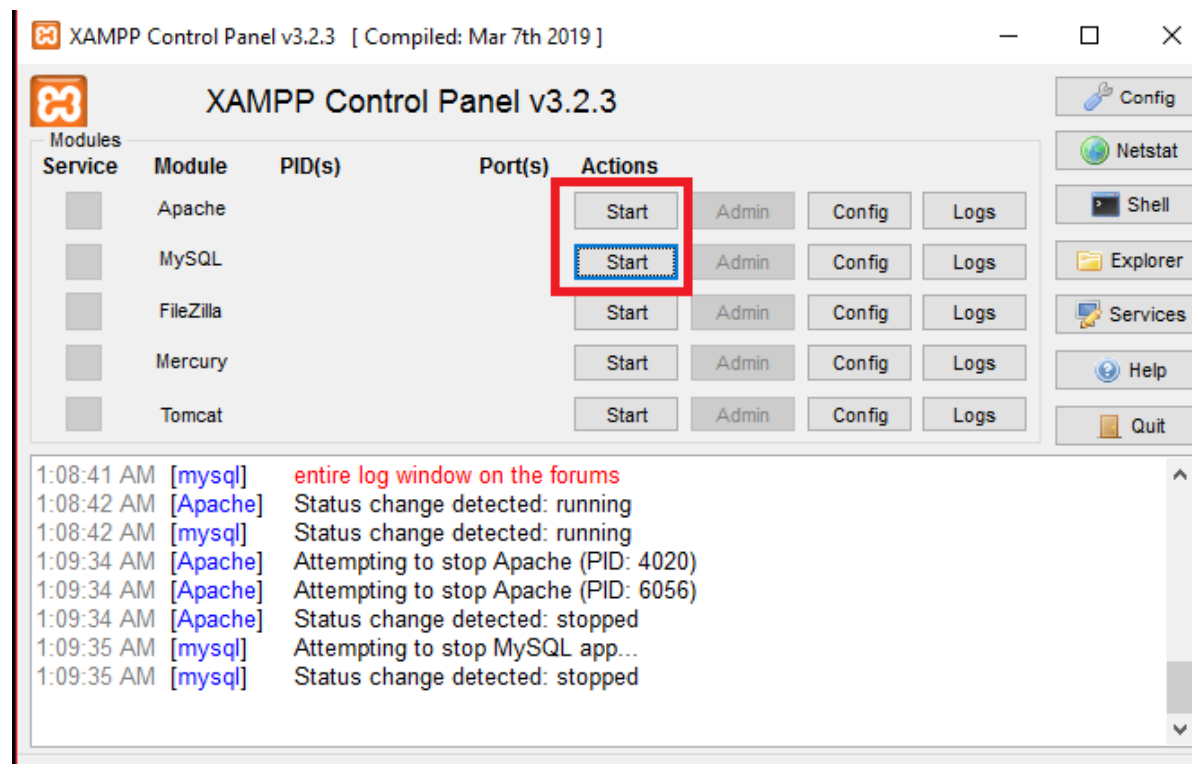
Tujuan praktikum ini adalah untuk mengetahui pengaruh indexing pada database. Langkah ke-1 sampai ke-11 hanya proses import database serta pengisian data. Proses perbandingan database sebelum dan sesudah indexing dimulai dari langkah ke-12.

Aplikasi/file yang dibutuhkan :

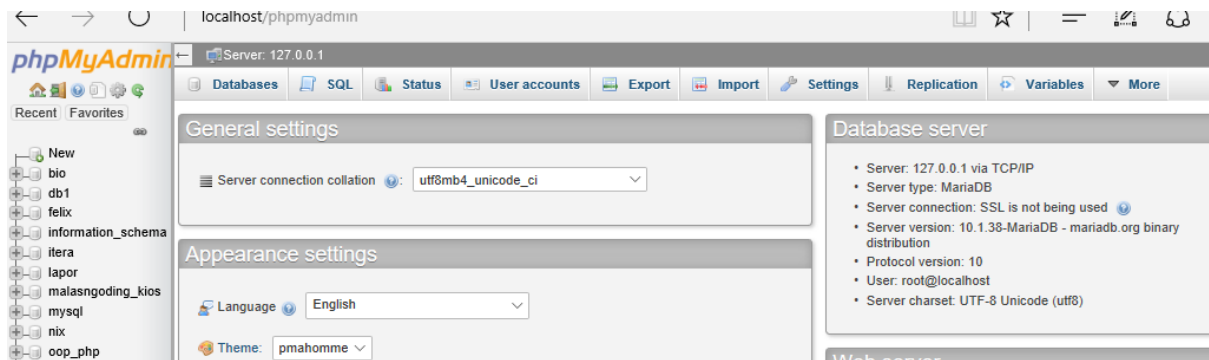
- XAMPP
- Web Browser
- File data.sql (unduh di bit.ly/3cAGgO2)

1. Lakukan instalasi software XAMPP di computer masing-masing

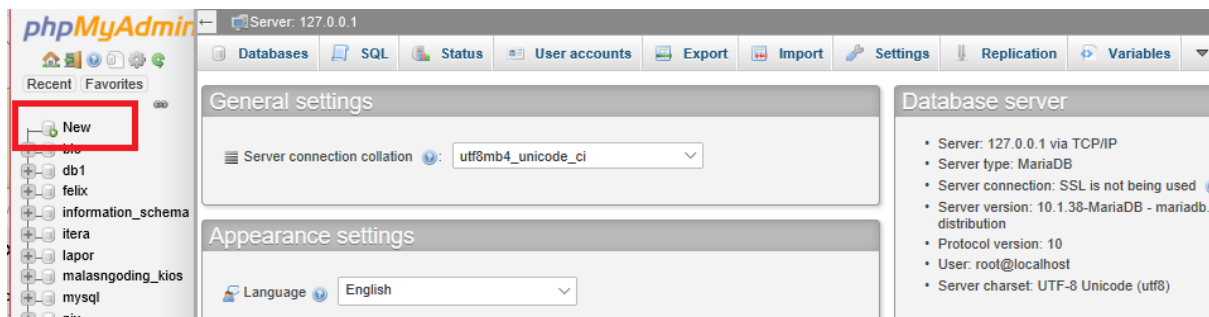
2. Buka aplikasi xampp, klik 'start' Apache dan MySQL.



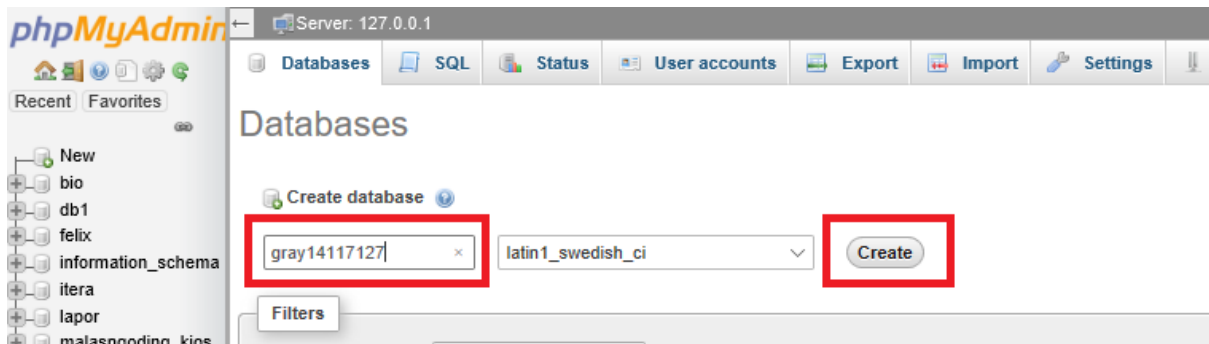
3. Buka browser, lalu ketik localhost/phpmyadmin. Maka akan terlihat halaman seperti gambar dibawah ini.



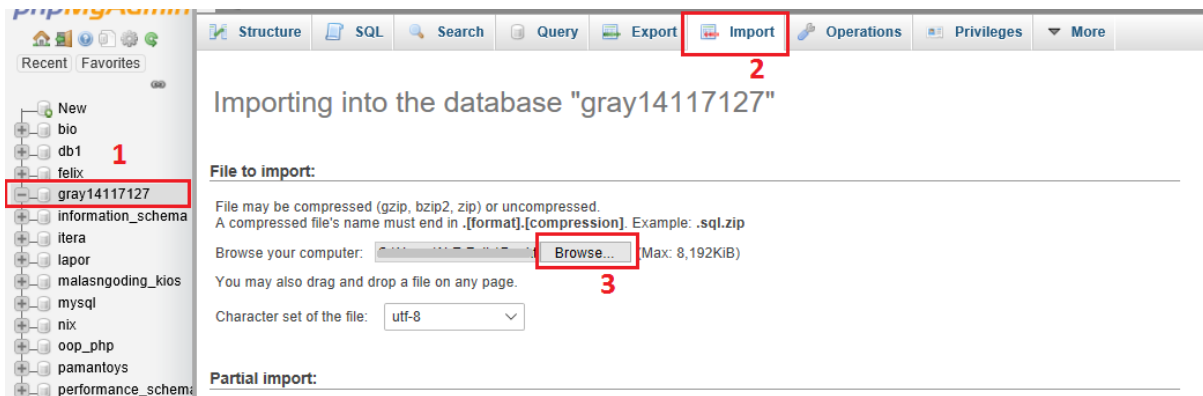
4. Klik 'new' untuk membuat database.



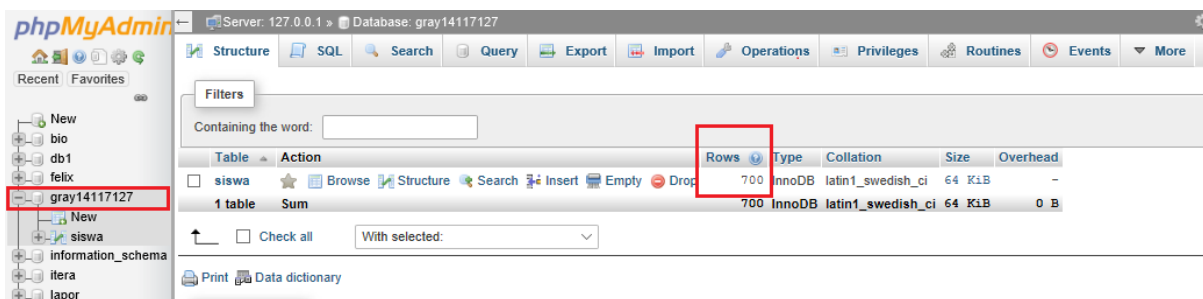
5. Beri nama database sesuai nama dan nim kalian (contoh : gray14117127), klik 'create' .



6. Klik database yang baru dibuat, pilih menu 'import', klik 'browse' lalu pilih file data.sql yang sudah di unduh. Scroll kebawah, klik 'go' untuk mengimport data.sql.



7. Setelah selesai, klik database , terlihat teabel baru 'siswa' yang memiliki 700 baris.

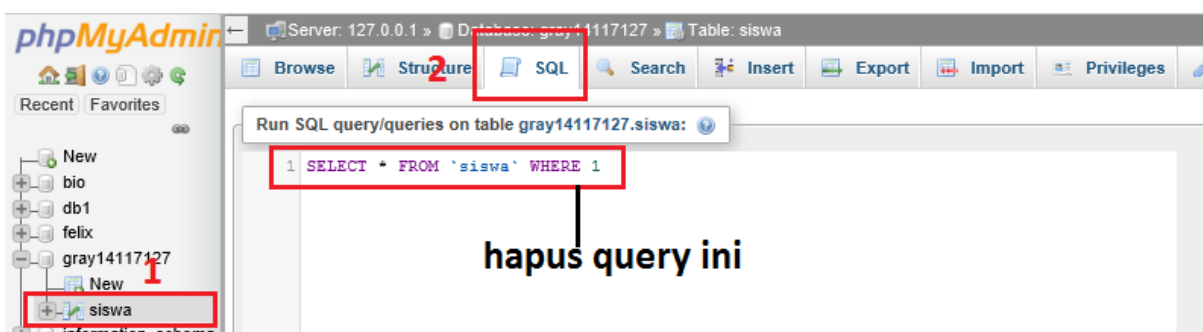


8. Selanjutnya isi/baris tabel 'siswa' akan diduplikasi sampai beberapa juta baris. Ini dilakukan supaya perbedaan yang terlihat lebih signifikan pada saat sesudah/sebelum indexing. Copy query ini :

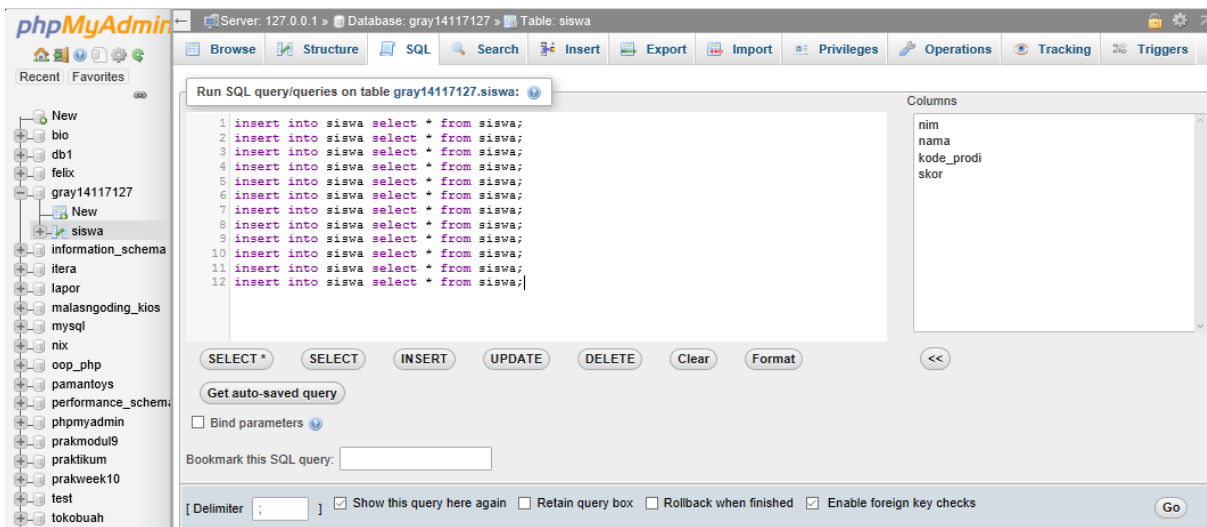
insert into siswa select * from siswa;

(ini adalah query untuk menduplikasi data dari tabel 'siswa' kedalam tabel itu sendiri).

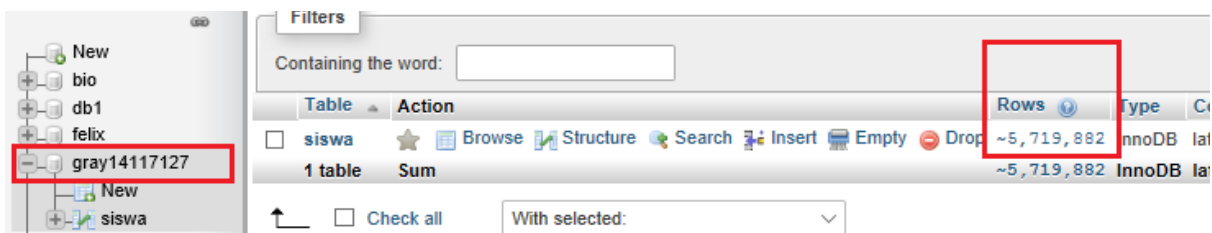
9. Klik tabel 'siswa', buka menu SQL, hapus query yang sudah ada.



10. Paste query (yang ada di langkah 8). Paste sebanyak 12 kali. Klik 'go' (proses ini membutuhkan waktu yang cukup lama karena kita akan menduplikasi data menjadi beberapa juta baris).



11. Untuk cek baris, klik database.



12. Selanjutnya akan dilakukan tes query. Kita akan menggunakan 2 query untuk melihat perbandingan sebelum indexing dan sesudah indexing di kolom 'nim'.

Query 1 : (query 'explain' akan menghasilkan penjelasan tentang query yang dieksekusi)

`explain select * from siswa`

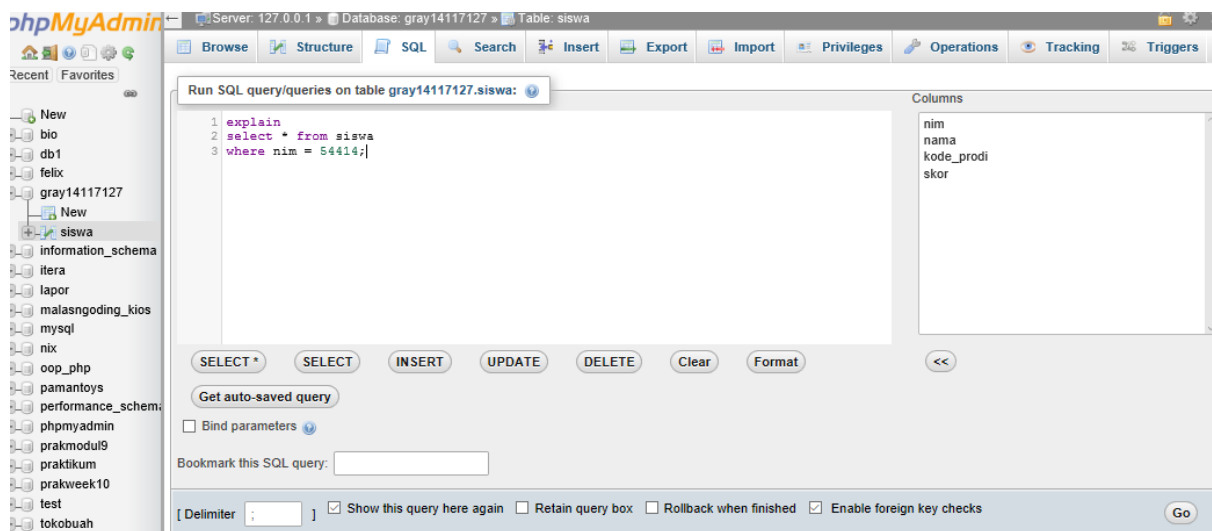
`where nim = 54414;`

Query 2 : (menghitung berapa baris yang memiliki nim 21212)

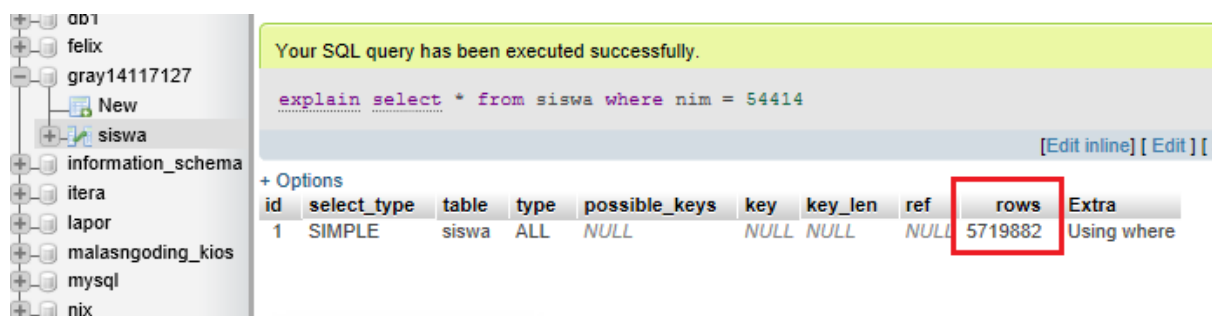
`select count(*) from siswa`

`where nim = 21212;`

13. Klik tabel 'siswa', ketik Query 1. Kemudian klik 'go'.

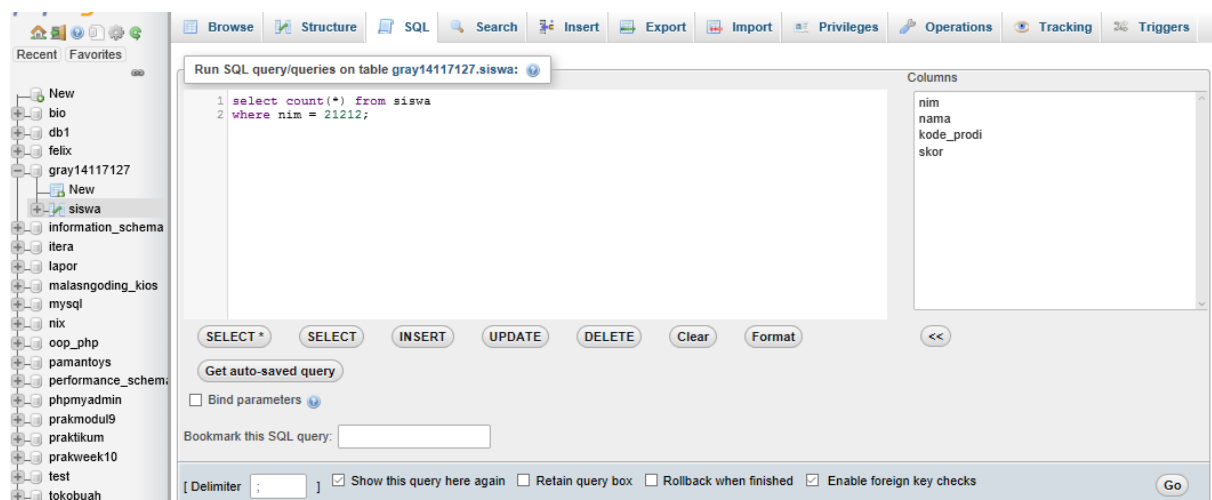


Hasil Query 1 :



Setelah proses selesai, perhatikan rows. Rows menunjukkan angka 5719882, berarti sistem membaca sebanyak 5719882 baris untuk menemukan atribut 'nim' yang berisi nilai 54414.

14. klik tabel 'siswa', ketik Query 2. Kemudian klik 'go'.



Hasil Query 2 :

The screenshot shows the MySQL interface. On the left, the database tree includes 'db1', 'felix', 'gray14117127', 'New', 'siswa', 'information_schema', 'itera', 'lapor', 'malasngoding_kios', 'mysql', 'nix', 'oop_php', and 'pamantoys'. The main window displays a query: `select count(*) from siswa where nim = 21212`. The result shows 'count(*)' with a value of 8192. A status bar at the top indicates 'Showing rows 0 - 0 (1 total, Query took 2.7724 seconds.)'. Below the query, there are controls for 'Show all', 'Number of rows' (set to 25), and a 'Filter rows' search box.

Waktu yang dibutuhkan untuk query ini adalah 2,7724 s.

15. Lakukan indexing pada kolom 'nim'.

Klik tabel siswa, klik structure, checkbox kolom 'nim', lalu klik index (proses membutuhkan waktu sedikit lama, bergantung jumlah baris dan spesifikasi PC).

The screenshot shows the MySQL Structure tool for the 'siswa' table. The table structure is displayed with columns: #1 nim (int(5)), #2 nama (varchar(35)), #3 kode_prodi (char(2)), and #4 skor (int(3)). The 'nim' column is selected with a checkbox. Below the table structure, there are buttons for 'Check all', 'With selected', 'Browse', 'Change', 'Drop', 'Primary', 'Unique', 'Index', and 'Fulltext'. The 'Index' button is highlighted with a red box and labeled '4'. The 'siswa' table is selected in the left sidebar, labeled '1'.

Setelah proses indexing selesai, akan terlihat daftar index (gambar di bawah ini menunjukkan bahwa kolom 'nim' sudah memiliki index).

The screenshot shows the MySQL Indexes tool. The 'Indexes' tab is selected, showing a list of indexes for the 'siswa' table. The index for the 'nim' column is highlighted with a red box. The index details are as follows:

Action	Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
Edit Drop	nim	BTREE	No	No	nim	1402	A	No	

16. Indexing sudah dilakukan. Sekarang lakukan tes query 1 dan query 2. (ulangi langkah 13 dan 14)

Hasil Query 1 :

Your SQL query has been executed successfully.

```
explain select * from siswa where nim = 54414
```

[Edit inline]

+ Options

id	select_type	table	type	possible_keys	key	key_len	ref	rows	Extra
1	SIMPLE	siswa	ref	nim	nim	4	const	8191	

Setelah indexing, sistem hanya membaca sebanyak 8191 baris (sebelum indexing : 5719882 baris).

Hasil Query 2 :

✓ Showing rows 0 - 0 (1 total, Query took 0.0697 seconds.)

```
select count(*) from siswa where nim = 21212
```

☐ Show all | Number of rows: 25 ▾ | Filter rows: Search this table

+ Options

count(*)
8192

Setelah indexing, waktu proses yang dibutuhkan hanya 0,0697 sekon (sebelum indexing : 2,7724 sekon).

C. Tugas

Lakukan tes query sebelum dan sesudah indexing pada kolom 'skor' (lakukan seperti langkah 13 sampai 16) dengan menggunakan query ini :

Query 1 :

```
explain select * from siswa  
where skor = 116;
```

Query 2 :

```
select count(*) from siswa  
where skor = 77;
```



REFERENSI

<https://pojokprogrammer.net/content/performance-tuning-sederhana-di-mysql-menggunakan-index>

<http://mysql.phi-integration.com/tips-mysql/copy-atau-duplikasi-table-pada-mysql>

