



MODUL PRAKTIKUM

SD3203-Teknologi Basis Data

**Program Studi Sains Data
Fakultas Sains
Institut Teknologi Sumatera**

2025

MODUL 3

Index Tunning

Index Tunning

1. Tujuan Praktikum

- a. Memahami Konsep Indexing: Mempelajari bagaimana indeks bekerja dalam database dan bagaimana mereka dapat meningkatkan kinerja query.
- b. Menerapkan Indexing pada Database: Mempraktikkan pembuatan, penggunaan, dan penghapusan indeks pada database MySQL menggunakan XAMPP.
- c. Mengoptimalkan Query dengan Index Tuning: Menganalisis dan mengoptimalkan query dengan menambahkan atau menghapus indeks yang sesuai.

2. Konsep Dasar

2.1 Pengertian Indeks

Indeks adalah struktur data yang digunakan dalam sistem manajemen basis data (DBMS) untuk mempercepat pencarian data dan meningkatkan kinerja query. Indeks bekerja seperti daftar isi dalam buku, di mana Anda dapat langsung menuju ke halaman tertentu tanpa harus membaca seluruh buku.

2.2 Komponen Indeks

- Kunci Indeks (Index Key): Nilai atau set nilai dari kolom yang diindeks. Indeks memetakan kunci ini ke lokasi data yang relevan dalam tabel.
- Pointer: Referensi atau alamat ke baris data yang terkait dalam tabel. Ketika DBMS menemukan kunci dalam indeks, pointer ini digunakan untuk mengakses baris yang sesuai di tabel.

2.3 Dua tipe utama Indeks

- Indeks Berurutan (Ordered Indices): Kunci pencarian disimpan dalam urutan tertentu, biasanya secara ascending atau descending.
- Indeks Hash (Hash Indices): Menggunakan fungsi hash untuk memetakan kunci pencarian ke lokasi tertentu yang disebut bucket. Cocok untuk pencarian tepat (exact match).

2.4 Manfaat Indeks

- Mempercepat Akses Data: Indeks memungkinkan DBMS untuk menemukan data lebih cepat tanpa harus memindai seluruh tabel.
- Mempercepat Operasi Query: Operasi seperti pencarian, pengurutan, dan pengelompokan data dapat dilakukan lebih cepat dengan adanya indeks.

- Mengurangi Beban I/O: Dengan menggunakan indeks, DBMS dapat mengurangi jumlah operasi input/output (I/O) yang diperlukan untuk mengambil data.

2.5 Jenis Indeks dalam MySQL

- Indeks Clustered: Mengubah urutan fisik data dalam tabel sesuai dengan urutan indeks. Hanya satu indeks clustered yang dapat dibuat per tabel.
- Indeks Nonclustered: Tidak mengubah urutan fisik data dalam tabel, tetapi membuat struktur terpisah yang menyimpan referensi ke baris data.

2.6 Kapan Indeks Dibutuhkan

- Klausula WHERE: Ketika kolom yang diindeks digunakan dalam klausula WHERE sebagai kriteria pencarian.
- Klausula ORDER BY dan GROUP BY: Ketika kolom yang diindeks digunakan dalam klausula ORDER BY atau GROUP BY.
- Fungsi Agregat: Ketika fungsi agregat seperti MAX dan MIN digunakan pada kolom yang diindeks.
- Join: Ketika kolom yang diindeks digunakan dalam operasi JOIN.

2.7 Kapan Indeks Tidak Dibutuhkan

- Tabel Kecil: Pada tabel dengan jumlah baris yang sedikit, indeks mungkin tidak memberikan manfaat yang signifikan.
- Kolom dengan Kardinalitas Rendah: Kolom dengan nilai yang sering berulang (misalnya, jenis kelamin) mungkin tidak memerlukan indeks.
- Operasi Write yang Sering: Indeks dapat memperlambat operasi INSERT, UPDATE, dan DELETE karena indeks perlu diperbarui setiap kali data berubah.

2.8 Aturan Penggunaan Indeks

- Buat Indeks untuk Kolom yang Sering Dicari: Kolom yang sering digunakan dalam klausula WHERE, ORDER BY, atau GROUP BY sebaiknya diindeks.
- Hindari Indeks yang Berlebihan: Terlalu banyak indeks dapat memperlambat operasi write dan memakan ruang penyimpanan.
- Gunakan Indeks Unik untuk Kolom dengan Nilai Unik: Indeks unik memastikan bahwa tidak ada duplikasi nilai dalam kolom tersebut.

3. Persiapan dan Instalansi

▪ Download dan XAMPP

Unduh XAMPP dari situs resmi: <https://www.apachefriends.org>

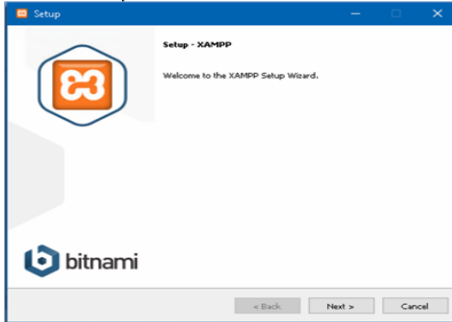
- Linux FAQs (https://www.apachefriends.org/faq_linux.html)
- Windows FAQs (https://www.apachefriends.org/faq_windows.html)
- OS X FAQs (https://www.apachefriends.org/faq_osx.html)



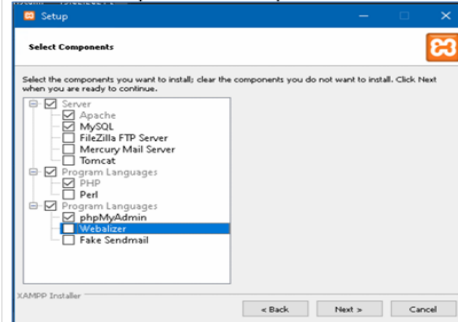
▪ Install XAMPP di Windows

- 3.1. Jalankan installer XAMPP yang telah diunduh.
- 3.2. Ikuti langkah-langkah instalasi dengan memilih komponen yang dibutuhkan (Apache, MySQL, dan phpMyAdmin).
- 3.3. Tentukan lokasi instalasi (direkomendasikan di C:\xampp untuk Windows).
- 3.4. Klik tombol **Install** dan tunggu hingga proses instalasi selesai.
- 3.5. Setelah instalasi selesai, jalankan XAMPP Control Panel.

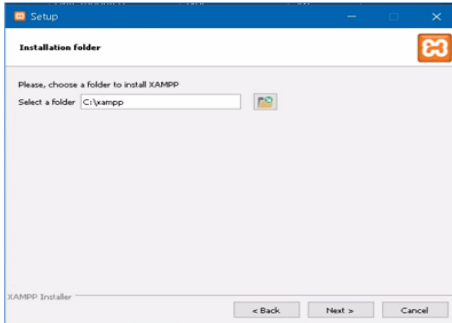
1. Start Setup



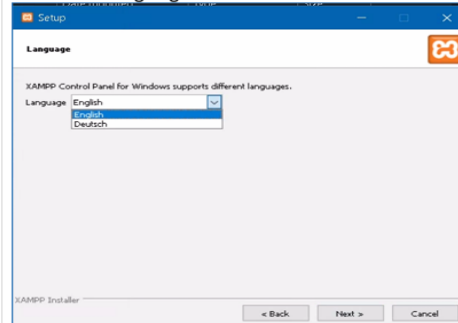
2. Select components as on picture



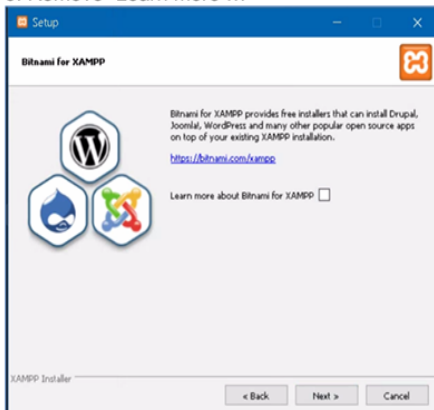
3. Select install folder



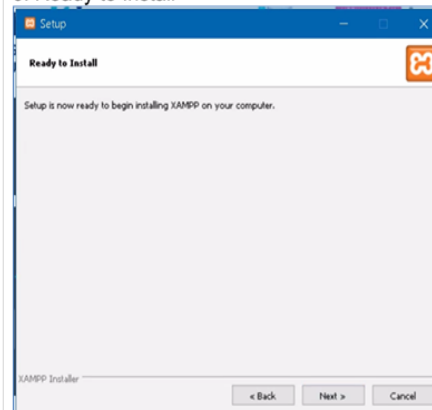
4. Select Language



5. Remove "Learn more ..."



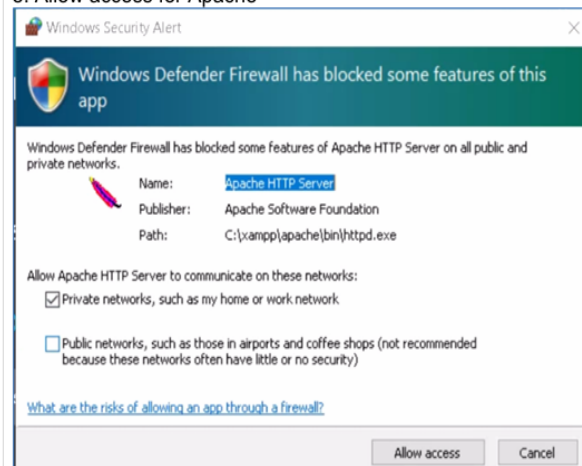
6. Ready to Install

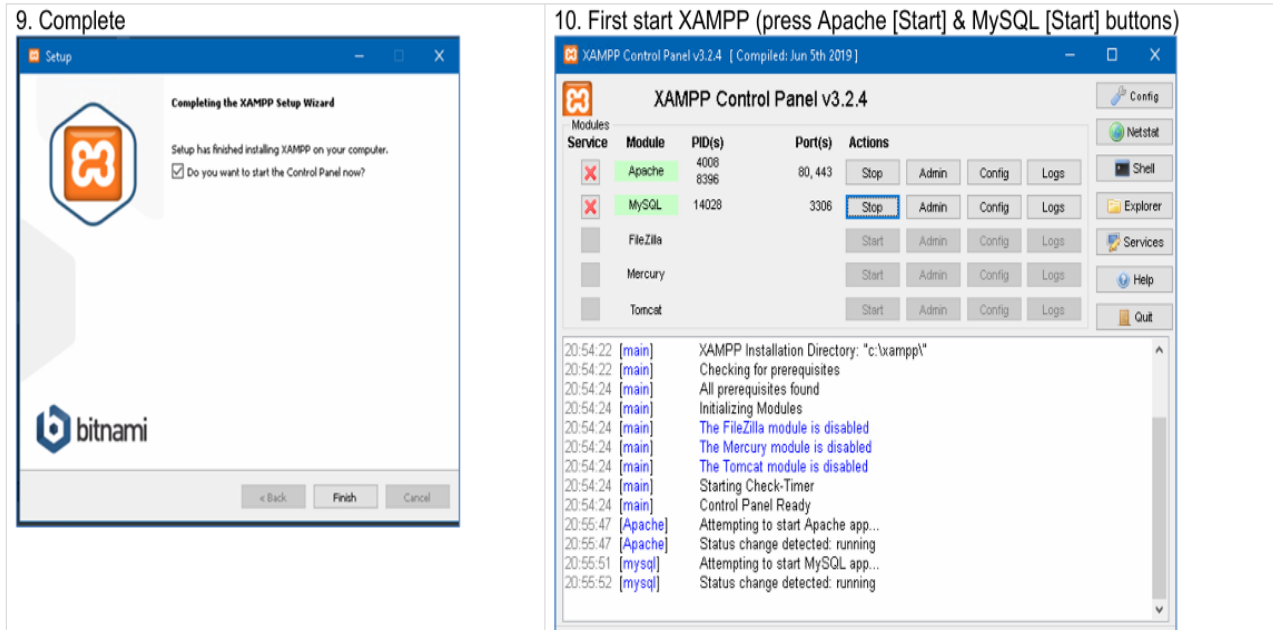


7. Installing



8. Allow access for Apache





4. Latihan Praktikum

Anda adalah seorang Database Administrator (DBA) di perusahaan XYZ Corp. Perusahaan ini memiliki database karyawan yang perlu dioptimasi untuk meningkatkan kinerja query. Anda diminta untuk melakukan serangkaian tugas terkait index tuning pada database karyawan.

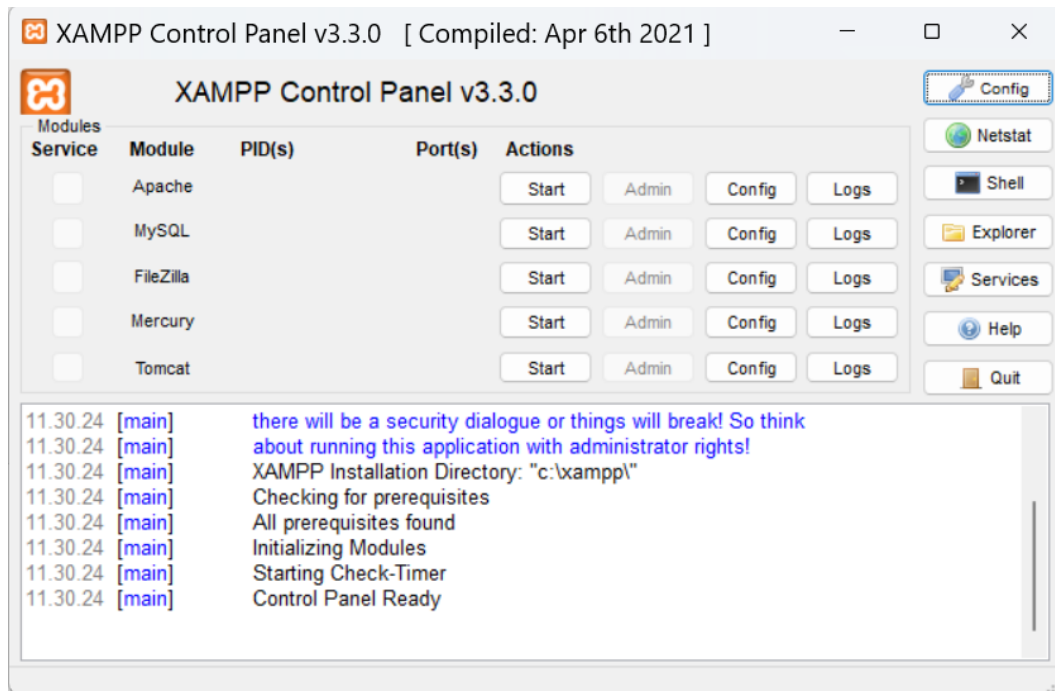
emp_id	emp_name	emp_SSN	emp_department	emp_salary	emp_join date
1	John Doe	123-45-6789	HR	50000.00	2020-01-15
2	Jane Smith	987-65-4321	IT	60000.00	2019-05-20
3	Alice Johnson	456-78-9123	Finance	55000.00	2021-03-10
4	Bob Brown	321-54-9876	IT	62000.00	2018-11-30

Berikut adalah langkah-langkah yang harus Anda lakukan:

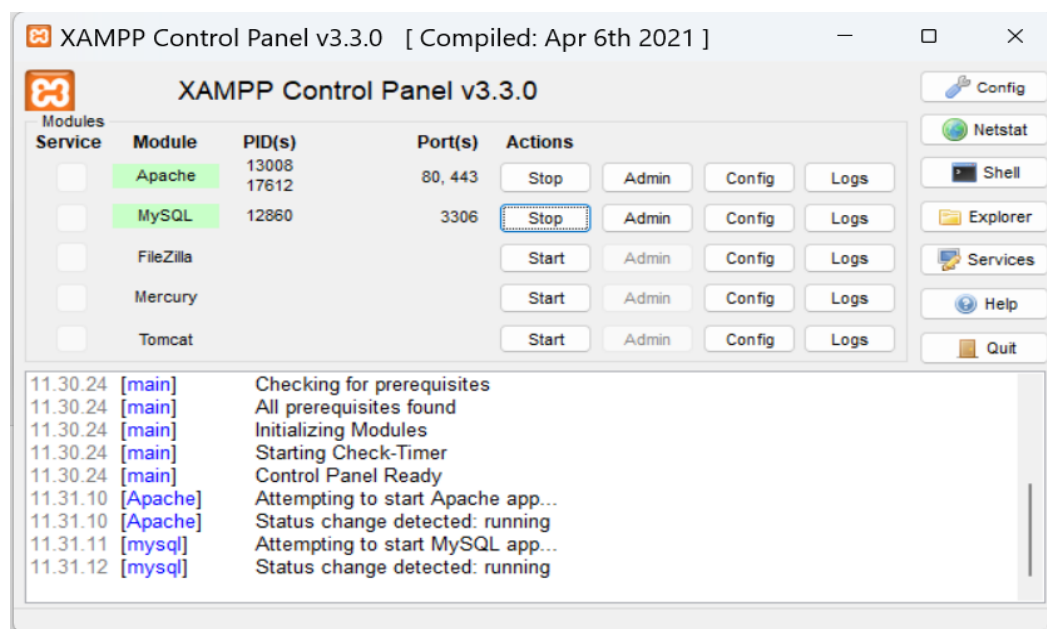
4.1 Persiapan Database

▪ Langkah 1 Menjalankan Apache dan MySQL di XAMPP

1. Buka **XAMPP Control Panel**.
2. Klik tombol **Start** pada layanan **Apache** dan **MySQL**.

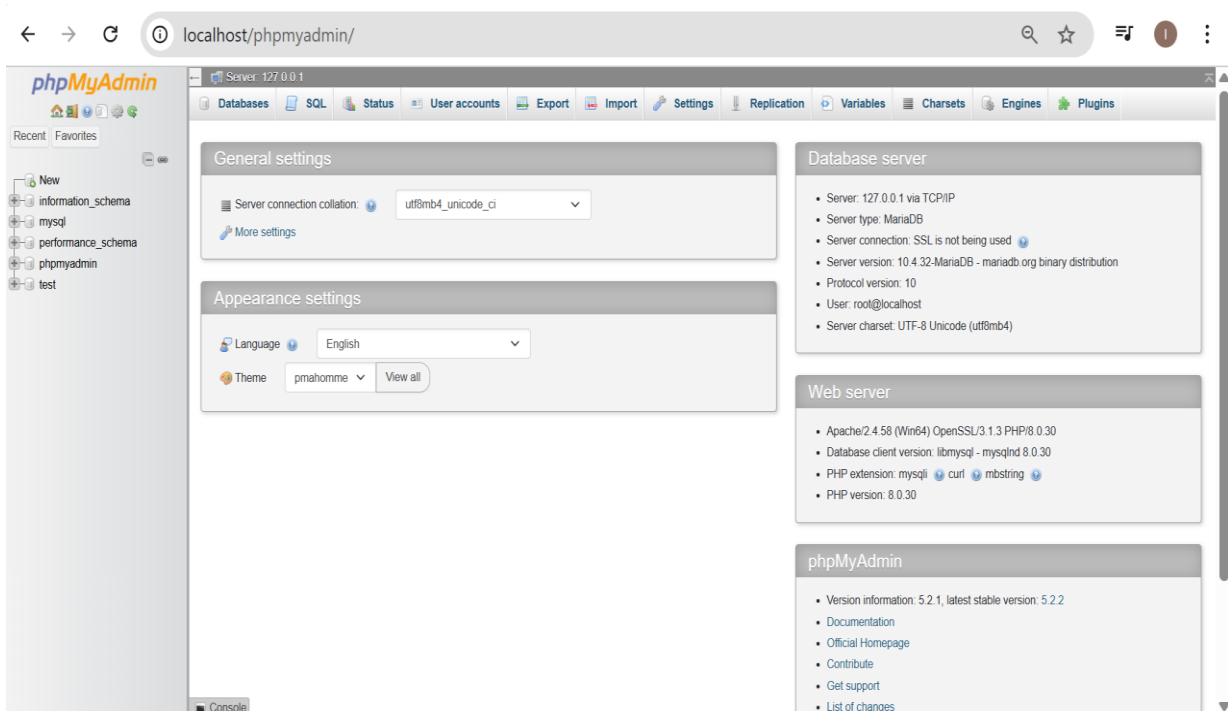


3. Pastikan kedua layanan tersebut berubah status menjadi "Running" dan berubah warna "hijau"



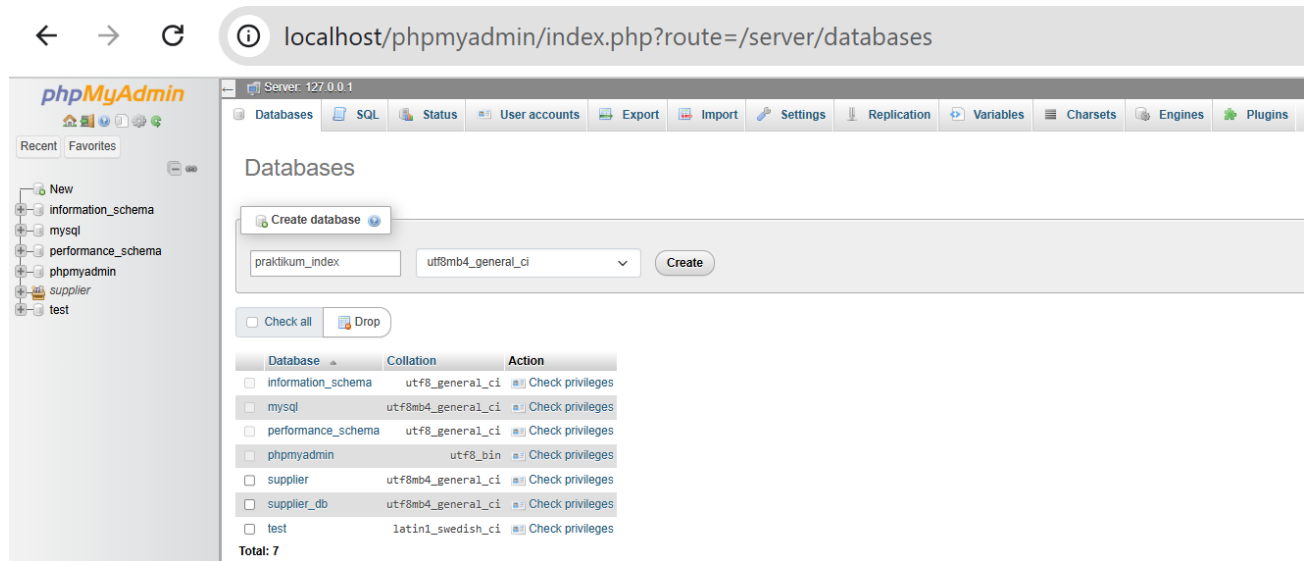
▪ Langkah 2 Mengakses phpMyAdmin

1. Buka browser web (Chrome, Firefox, Edge, dll.).
2. Ketik `http://localhost/phpmyadmin` di address bar dan tekan Enter.
3. Halaman utama phpMyAdmin akan muncul, menampilkan daftar database yang ada.

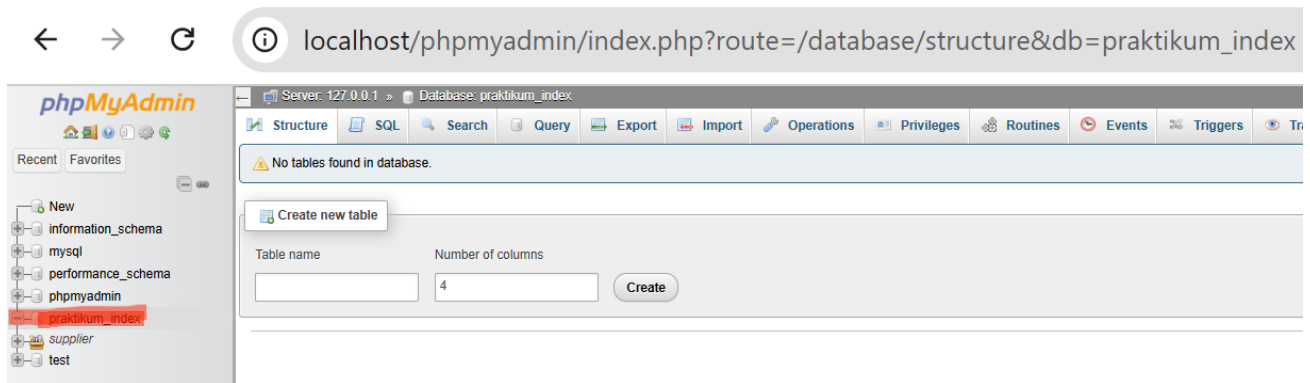


▪ Langkah 3 Membuat Database Baru di phpMyAdmin

1. Di halaman utama phpMyAdmin, klik tab **Database**.
2. Masukkan nama database: **praktikum_index** pada kolom "Create database".
3. Klik tombol **Create** untuk membuat database.



Setelah database berhasil dibuat, klik database **praktikum_index** dari daftar di sebelah kiri



Langkah 4 Buat Tabel

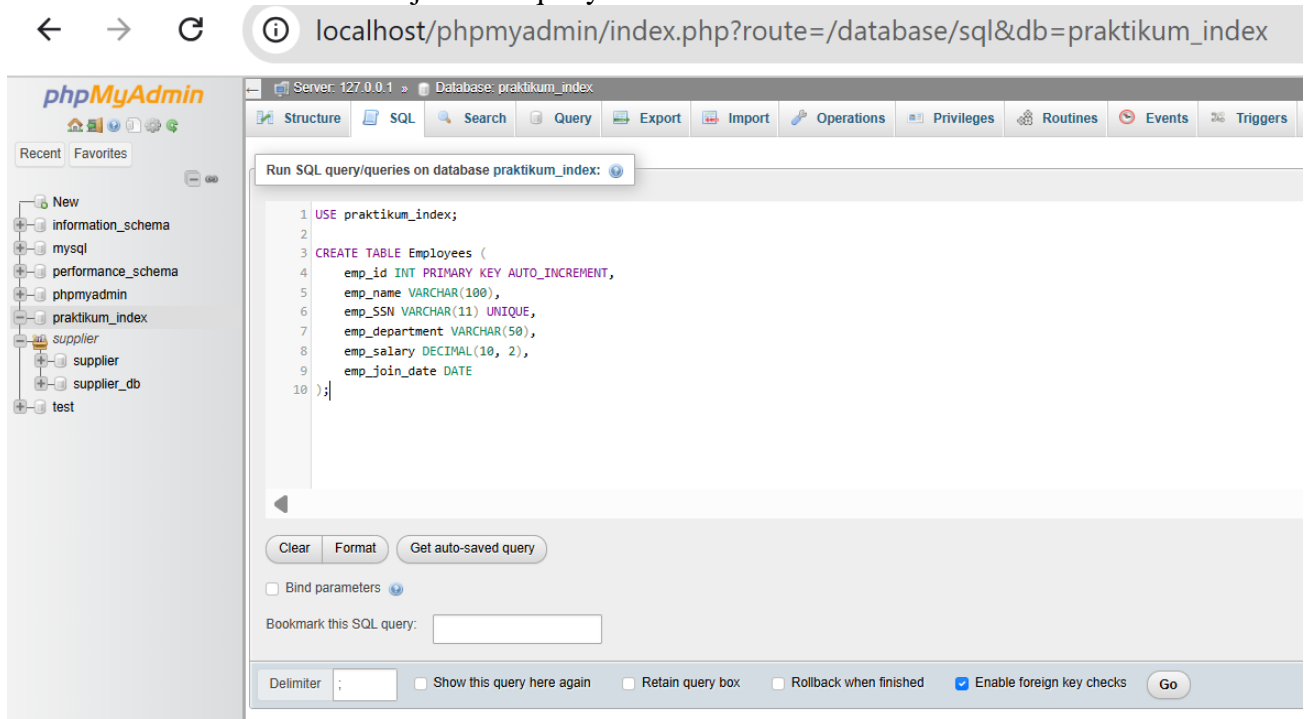
- Buat tabel Employees dengan struktur berikut:

 1. Pilih database **praktikum_index** dari side bar kiri.
 2. Klik tab **SQL** di bagian atas.
 3. Masukkan query berikut ke dalam kotak SQL:

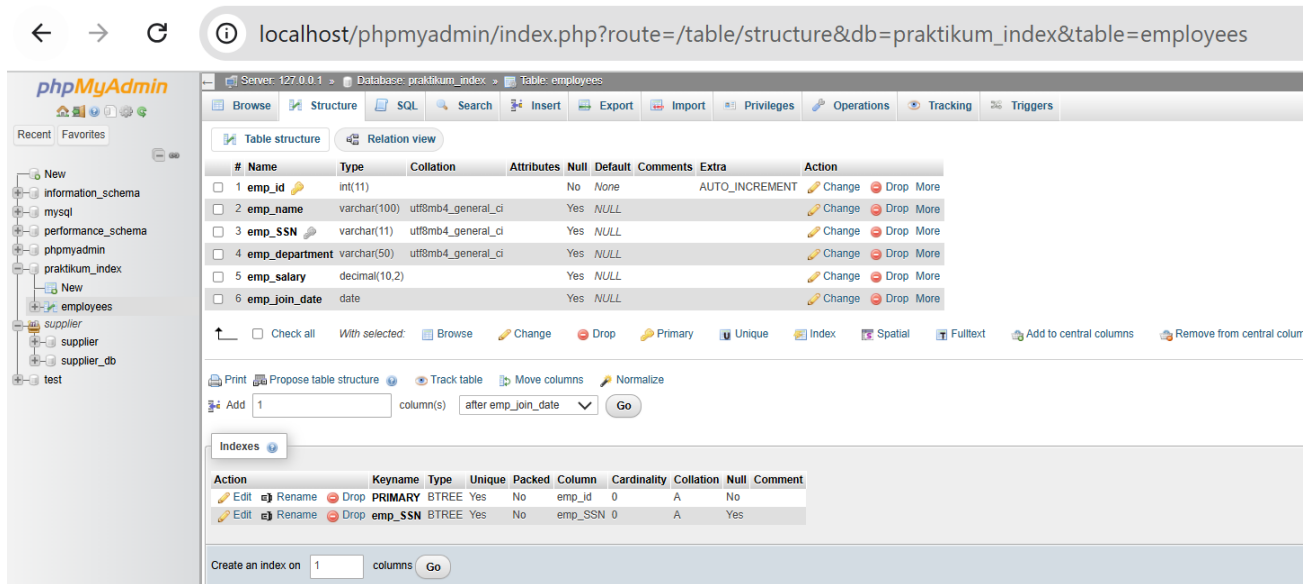
```
USE praktikum_index;

CREATE TABLE Employees (
  emp_id INT PRIMARY KEY AUTO_INCREMENT,
  emp_name VARCHAR(100),
  emp_SSN VARCHAR(11) UNIQUE,
  emp_department VARCHAR(50),
  emp_salary DECIMAL(10, 2),
  emp_join_date DATE
);
```

4. Klik tombol **Go** untuk menjalankan query.



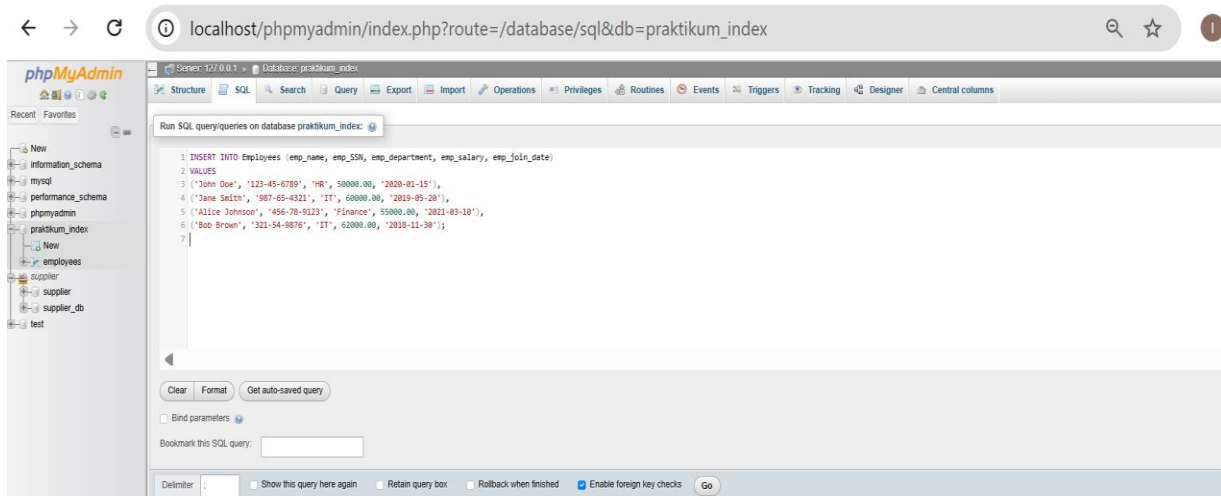
Maka bentuk struktur tabel Employees sebagai berikut.



- **Langkah 5 Memasukkan Data ke dalam Tabel**
 - Masukkan data ke tabel Employees sebagai berikut:
 - 1. Klik database **praktikum_index** di side bar kiri.
 - 2. Klik tab **SQL** di bagian atas.
 - 3. Masukkan query berikut ke dalam kotak SQL:

```
INSERT INTO Employees (emp_name, emp_SSN, emp_department, emp_salary,
emp_join_date)
VALUES
('John Doe', '123-45-6789', 'HR', 50000.00, '2020-01-15'),
('Jane Smith', '987-65-4321', 'IT', 60000.00, '2019-05-20'),
('Alice Johnson', '456-78-9123', 'Finance', 55000.00, '2021-03-10'),
('Bob Brown', '321-54-9876', 'IT', 62000.00, '2018-11-30');
```

- 4. Klik **Go** untuk menjalankan query.



4.2 Membuat dan Menggunakan Indeks

Indeks dibuat untuk mempercepat pencarian data pada kolom tertentu. Berikut adalah langkah-langkah untuk membuat indeks:

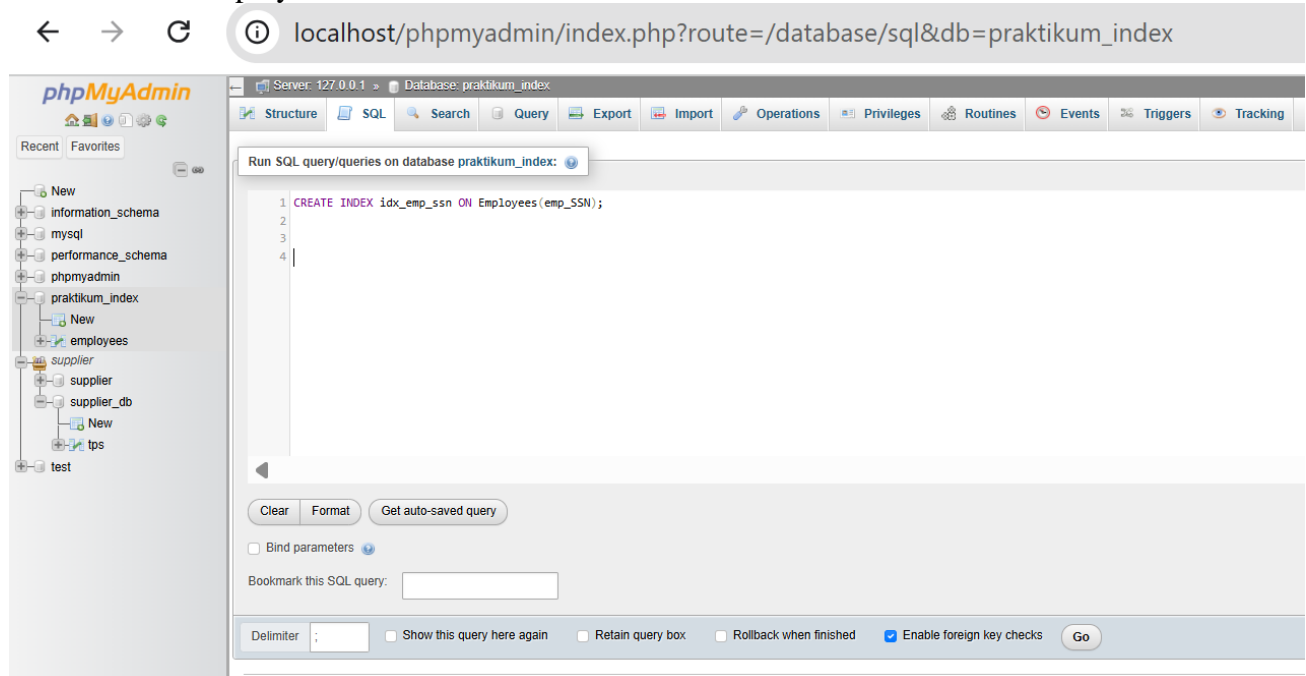
a. Membuat Indeks

▪ **Langkah 1 Buat Indeks pada Kolom emp_SSN:**

- Kolom emp_SSN sudah memiliki constraint UNIQUE, yang secara otomatis membuat indeks unik. Namun, kita bisa membuat indeks secara manual:

```
CREATE INDEX idx_emp_ssn ON Employees(emp_SSN);
```

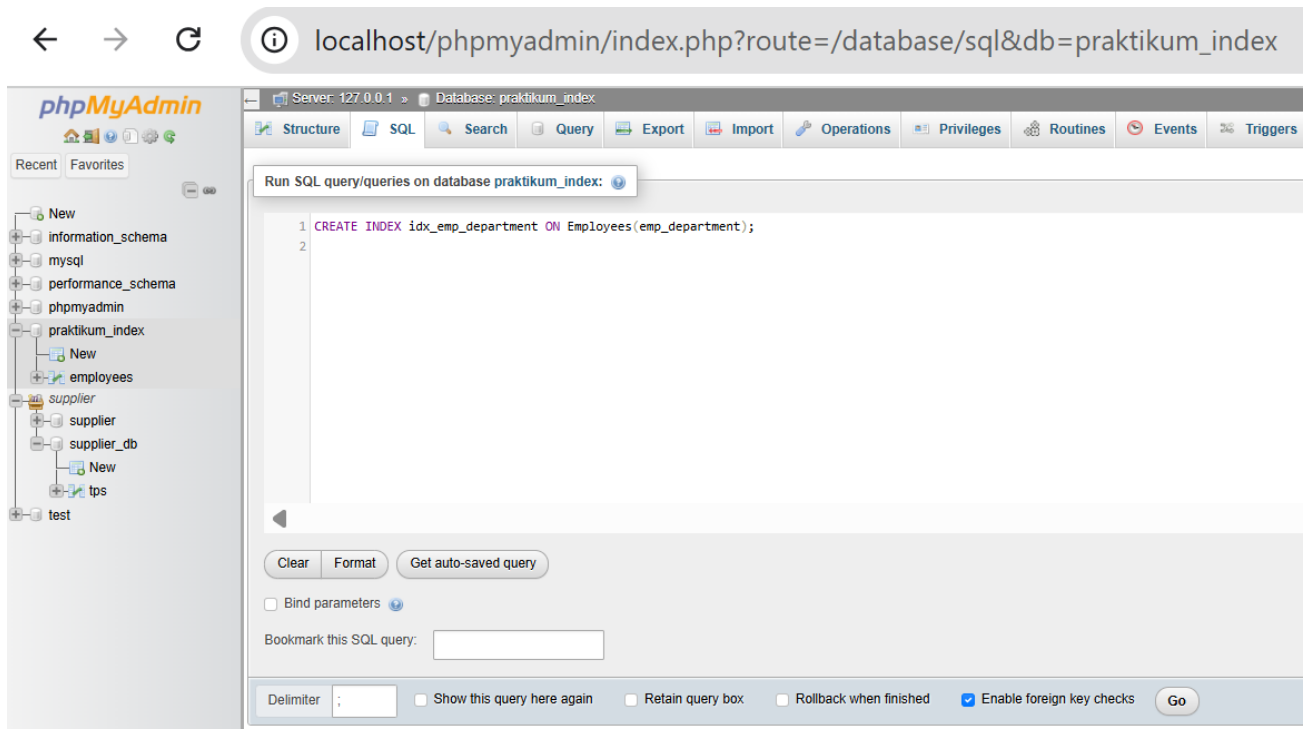
- ✓ CREATE INDEX adalah perintah untuk membuat indeks.
- ✓ idx_emp_ssn adalah nama indeks yang kita buat.
- ✓ Employees(emp_SSN) menunjukkan bahwa indeks dibuat pada kolom emp_SSN di tabel Employees.



▪ **Langkah 2 Buat Indeks pada Kolom yang Sering Digunakan dalam Klausa WHERE:**

- Buat indeks non-clustered pada kolom emp_department untuk mempercepat pencarian berdasarkan departemen:

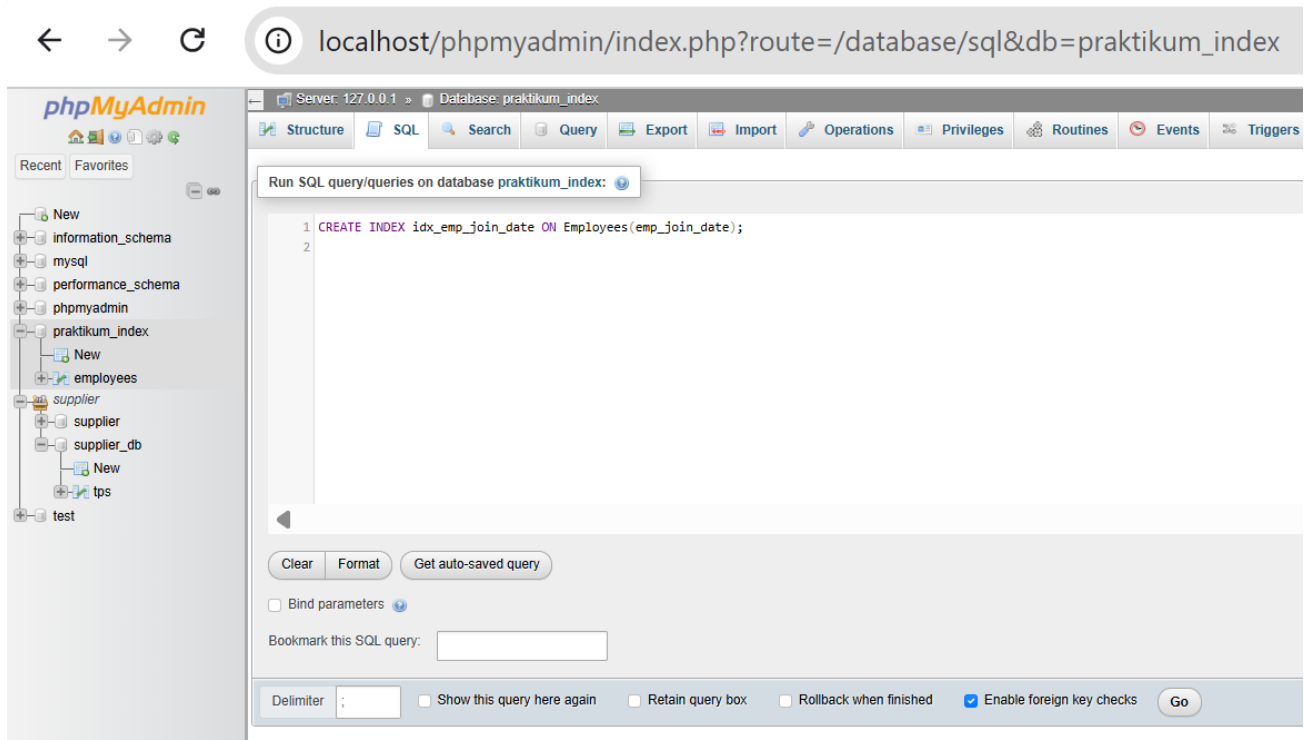
```
CREATE INDEX idx_emp_department ON Employees(emp_department);
```



▪ **Langkah 3 Buat Indeks pada Kolom yang Digunakan dalam Klausa ORDER BY:**

- Buat indeks pada kolom `emp_join_date` untuk mempercepat query yang melibatkan rentang tanggal:

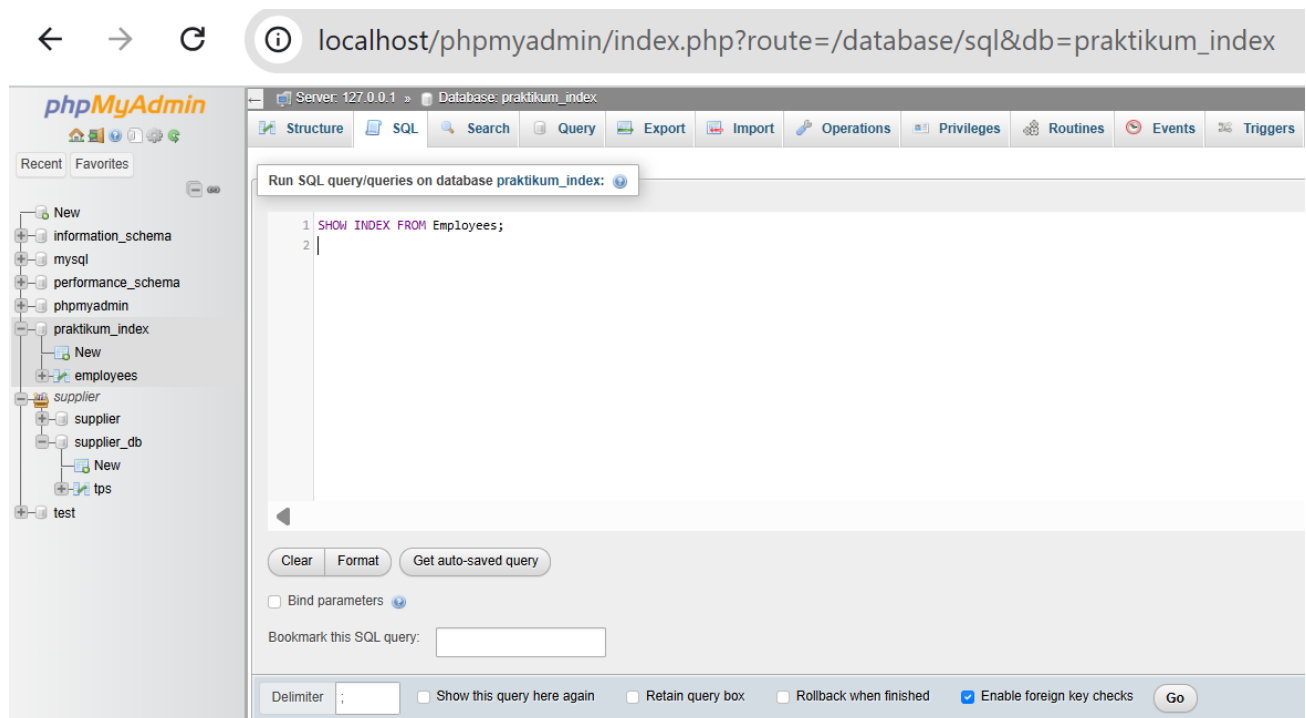
```
CREATE INDEX idx_emp_join_date ON Employees(emp_join_date);
```



▪ **Langkah 4 Verifikasi Indeks:**

- Untuk melihat indeks yang telah dibuat, jalankan query berikut:

```
SHOW INDEX FROM Employees;
```

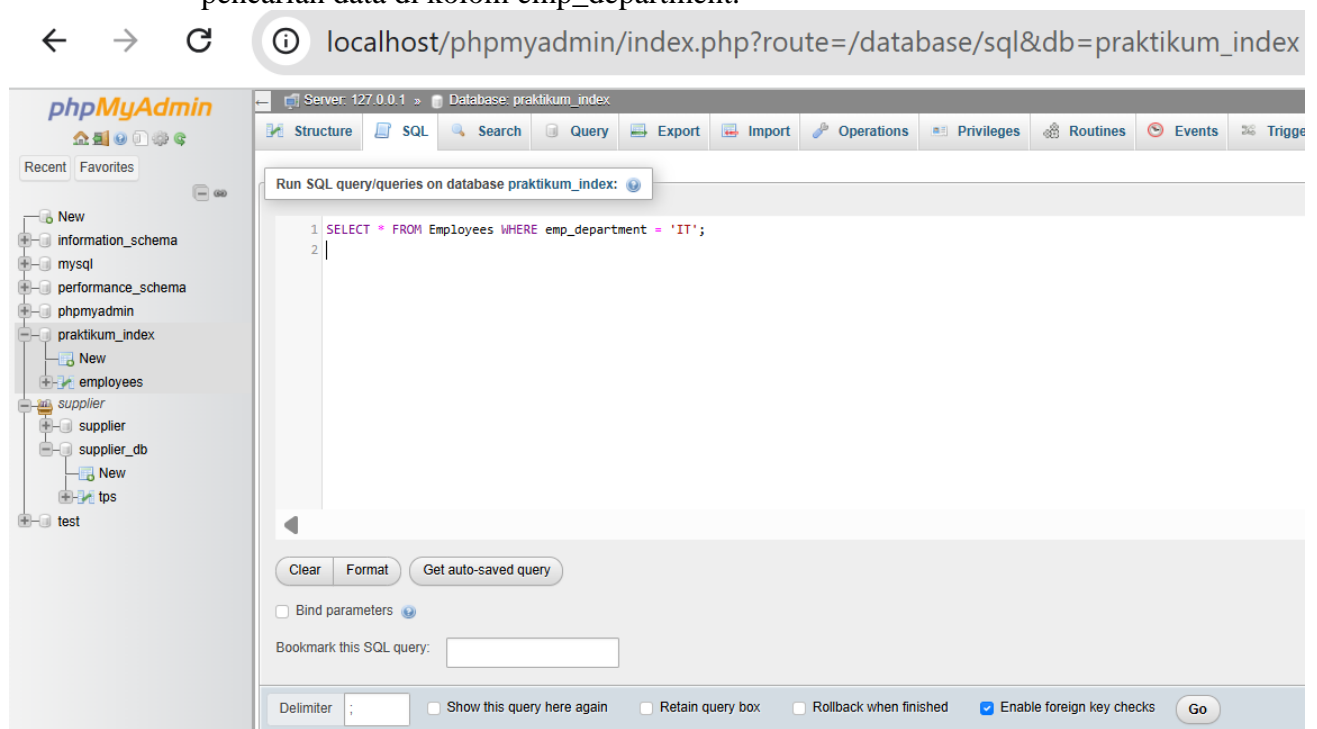
b. Menggunakan Indeks

Setelah indeks dibuat, DBMS akan secara otomatis menggunakan indeks tersebut untuk mempercepat query yang melibatkan kolom yang diindeks.

▪ Langkah 1 Query dengan Klausa WHERE:

```
SELECT * FROM Employees WHERE emp_department = 'IT';
```

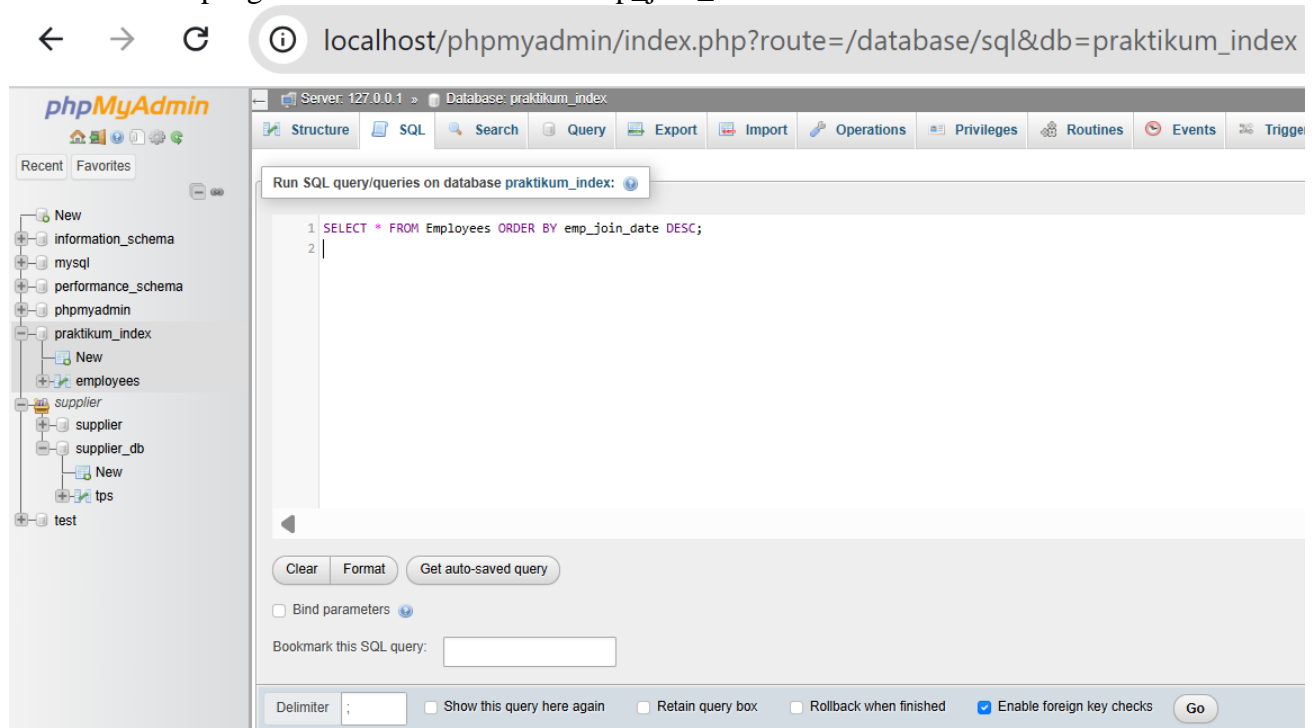
- DBMS akan menggunakan indeks idx_emp_department untuk mempercepat pencarian data di kolom emp_department.



▪ Langkah 2 Query dengan Klausa ORDER BY:

```
SELECT * FROM Employees ORDER BY emp_join_date DESC;
```

- DBMS akan menggunakan indeks idx_emp_join_date untuk mempercepat pengurutan data berdasarkan emp_join_date.

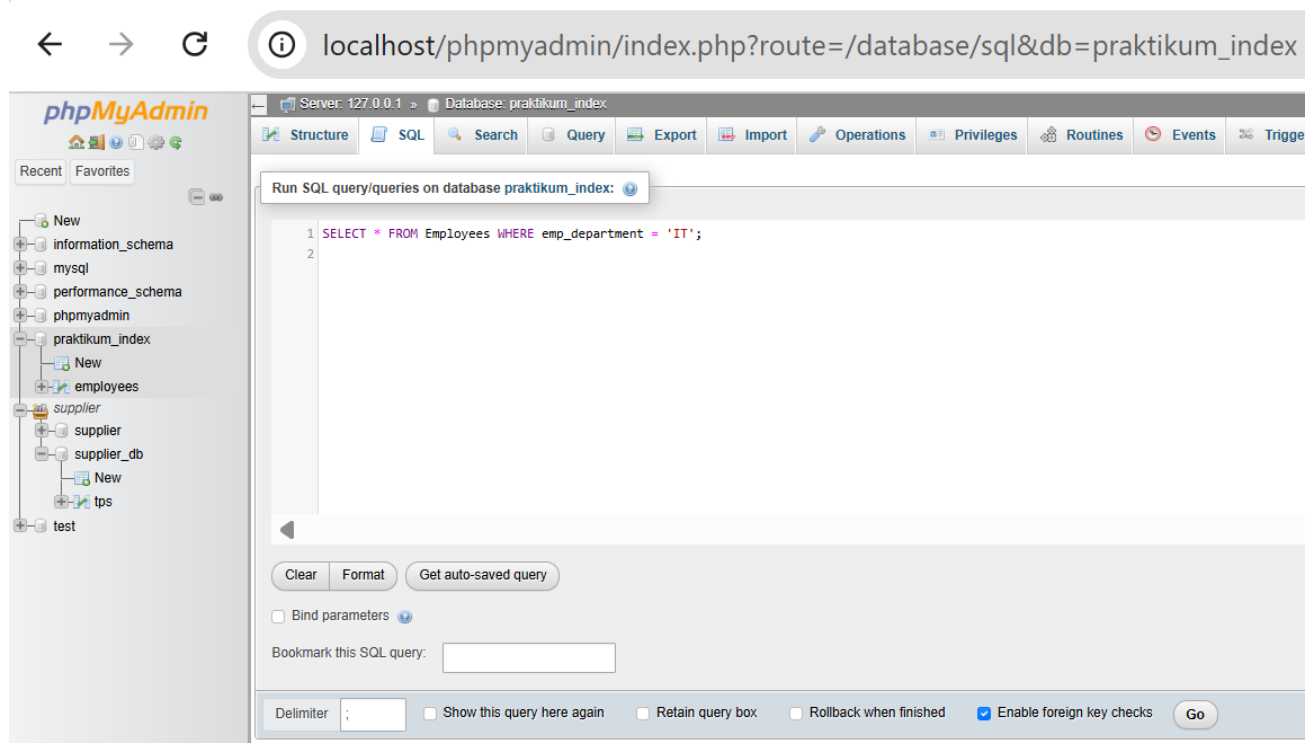


4.3 Mengoptimalkan Query dengan Indeks

▪ Langkah 1 Query Tanpa Indeks:

- Jalankan query berikut tanpa indeks dan catat waktu eksekusi:

```
SELECT * FROM Employees WHERE emp_department = 'IT';
```



- Catat waktu eksekusi query ini.

← → ↻ ⓘ localhost/phpmyadmin/index.php?route=/sql&db=praktikum_index&table=Employees&pos=0

phpMyAdmin

Recent Favorites

Server: 127.0.0.1 » Database: praktikum_index » Table: Employees

Showing rows 0 - 3 (4 total, Query took 0.0005 seconds.)

`SELECT * FROM `Employees``

Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

Show all Number of rows: 25 Filter rows: Search this table Sort by key: None

Extra options

	emp_id	emp_name	emp_SSN	emp_department	emp_salary	emp_join_date
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	1	John Doe	123-45-6789	HR	50000.00	2020-01-15
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	2	Jane Smith	987-65-4321	IT	60000.00	2019-05-20
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	3	Alice Johnson	456-78-9123	Finance	55000.00	2021-03-10
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	4	Bob Brown	321-54-9876	IT	62000.00	2018-11-30

Check all With selected: Edit Copy Delete Export

Show all Number of rows: 25 Filter rows: Search this table Sort by key: None

Query results operations

Print Copy to clipboard Export Display chart Create view

Bookmark this SQL query

▪ **Langkah 2 Query dengan Indeks:**

- Jalankan query yang sama setelah membuat indeks pada kolom emp_department:

```
SELECT * FROM Employees WHERE emp_department = 'IT';
```

← → ↻ ⓘ localhost/phpmyadmin/index.php?route=/database/sql&db=praktikum_index

phpMyAdmin

Recent Favorites

Server: 127.0.0.1 » Database: praktikum_index

Structure SQL Search Query Export Import Operations Privileges Routines Events Triggers

Run SQL query/queries on database praktikum_index:

```
1 SELECT * FROM Employees WHERE emp_department = 'IT';
2
```

Clear Format Get auto-saved query

Bind parameters ⓘ

Bookmark this SQL query:

Delimiter ; Show this query here again Retain query box Rollback when finished ☒ Enable foreign key checks Go

- Bandingkan waktu eksekusi sebelum dan setelah membuat indeks. Waktu eksekusi seharusnya lebih cepat setelah indeks dibuat.

← → ↻ localhost/phpmyadmin/index.php?route=/sql&db=praktikum_index&table=Employees&pos=0

phpMyAdmin

Recent Favorites

Server: 127.0.0.1 » Database: praktikum_index » Table: Employees

Browse Structure SQL Search Insert Export Import Privileges Operations Tracking Triggers

Showing rows 0 - 3 (4 total, Query took 0.0005 seconds)

SELECT * FROM `Employees`

Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

Show all Number of rows: 25 Filter rows: Search this table Sort by key: None

Extra options

	emp_id	emp_name	emp_SSN	emp_department	emp_salary	emp_join_date
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	1	John Doe	123-45-6789	HR	50000.00	2020-01-15
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	2	Jane Smith	987-65-4321	IT	60000.00	2019-05-20
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	3	Alice Johnson	456-78-9123	Finance	55000.00	2021-03-10
<input type="checkbox"/> Edit <input type="checkbox"/> Copy <input type="checkbox"/> Delete	4	Bob Brown	321-54-9876	IT	62000.00	2018-11-30

Check all With selected: Edit Copy Delete Export

Show all Number of rows: 25 Filter rows: Search this table Sort by key: None

Query results operations

Print Copy to clipboard Export Display chart Create view

Bookmark this SQL query

Label: ☐ Let every user access this bookmark

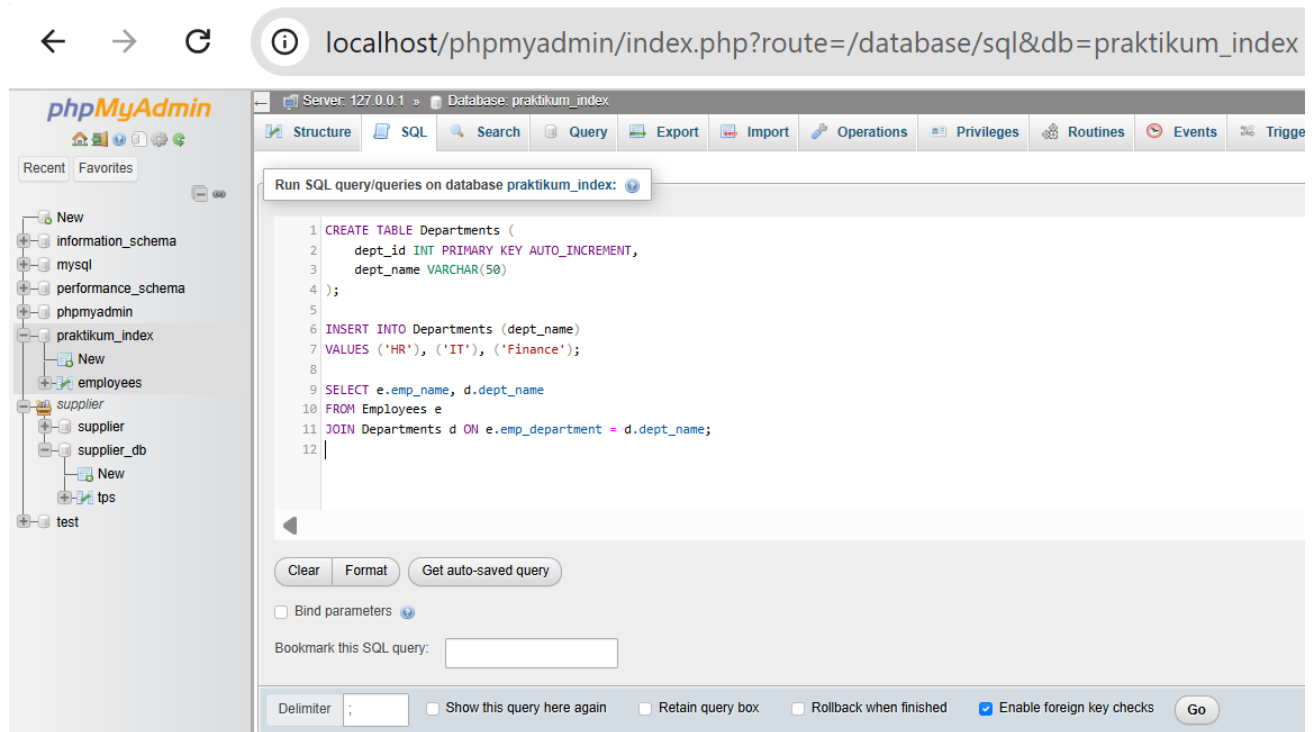
▪ Langkah 3 Optimalkan Query dengan Klausa JOIN:

- Buat tabel baru Departments dan lakukan JOIN dengan tabel Employees:

```
CREATE TABLE Departments (
    dept_id INT PRIMARY KEY AUTO_INCREMENT,
    dept_name VARCHAR(50)
);

INSERT INTO Departments (dept_name)
VALUES ('HR'), ('IT'), ('Finance');

SELECT e.emp_name, d.dept_name
FROM Employees e
JOIN Departments d ON e.emp_department = d.dept_name;
```



4.4 Menghapus Indeks yang Tidak Diperlukan

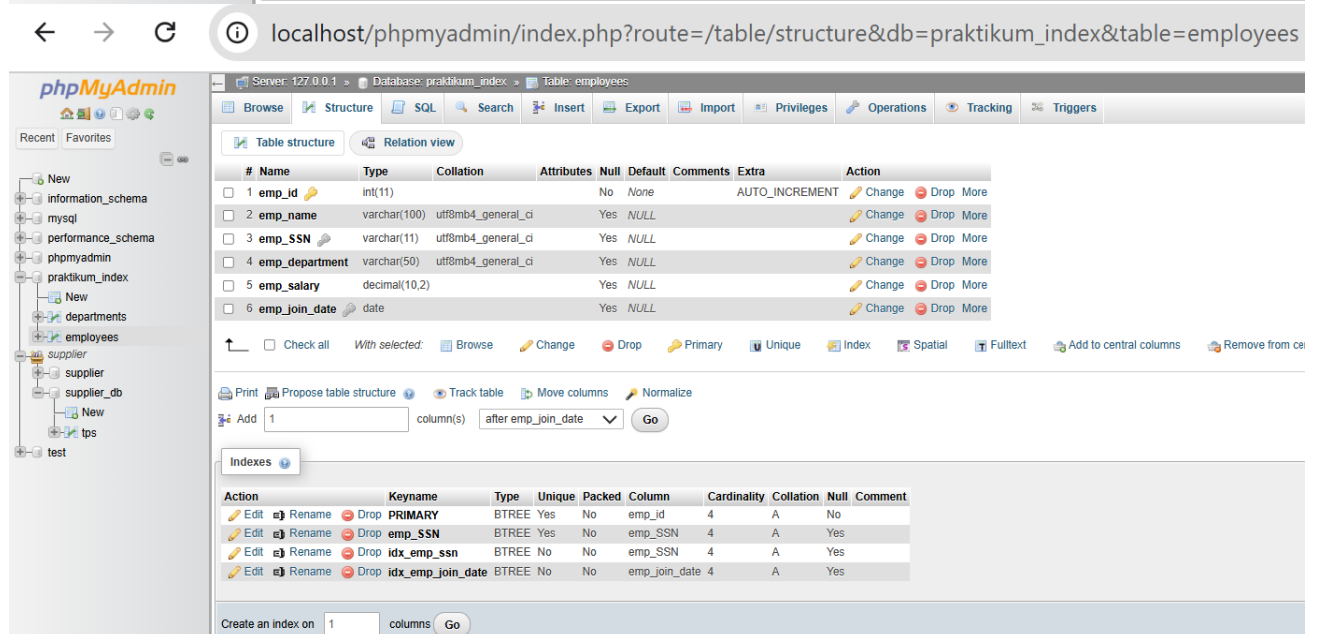
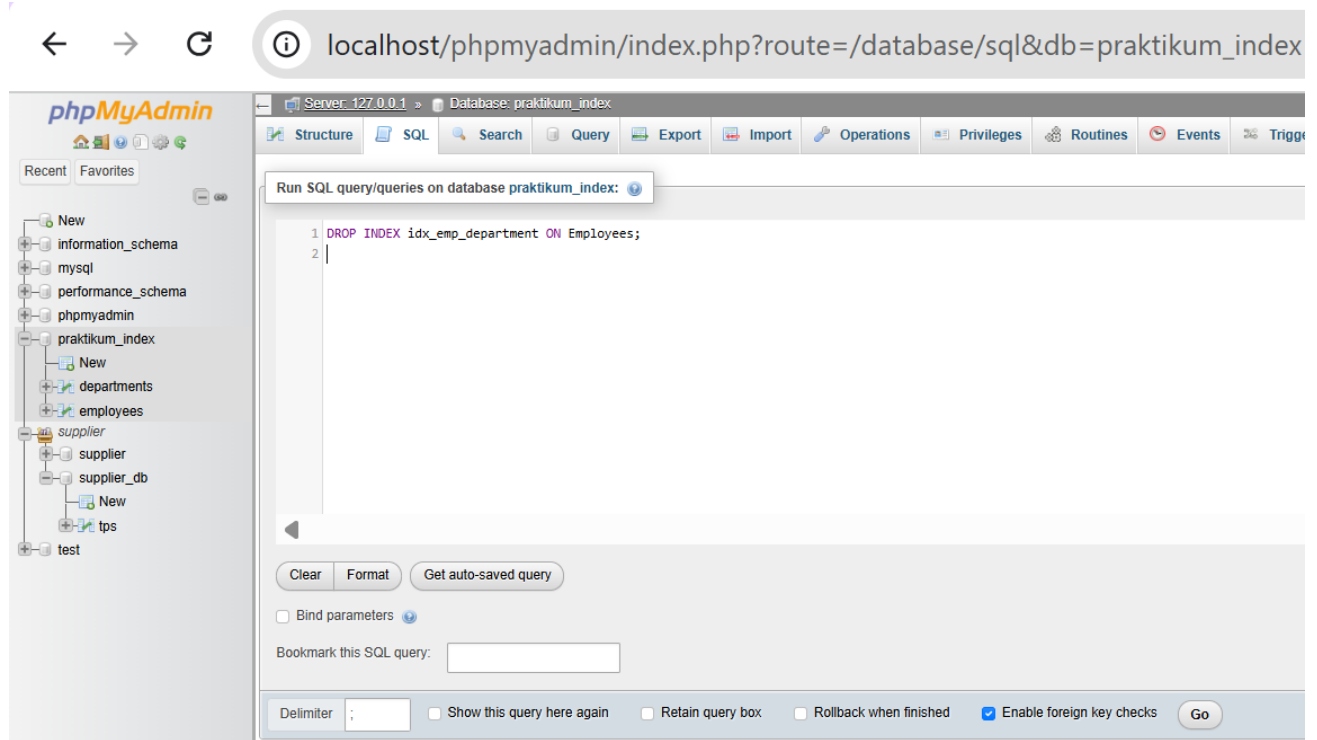
- **Indeks Tidak Digunakan:** Jika indeks tidak digunakan dalam query apa pun, sebaiknya dihapus untuk menghemat ruang penyimpanan.
- **Indeks Memperlambat Operasi Write:** Indeks dapat memperlambat operasi INSERT, UPDATE, dan DELETE karena indeks perlu diperbarui setiap kali data berubah.
- **Indeks Berlebihan:** Terlalu banyak indeks pada tabel dapat memperlambat kinerja database secara keseluruhan.

▪ Langkah 1 Hapus Indeks:

- Misalnya, kita ingin menghapus indeks idx_emp_department pada tabel Employees, Anda dapat menghapusnya:

```
DROP INDEX idx_emp_department ON Employees;
```

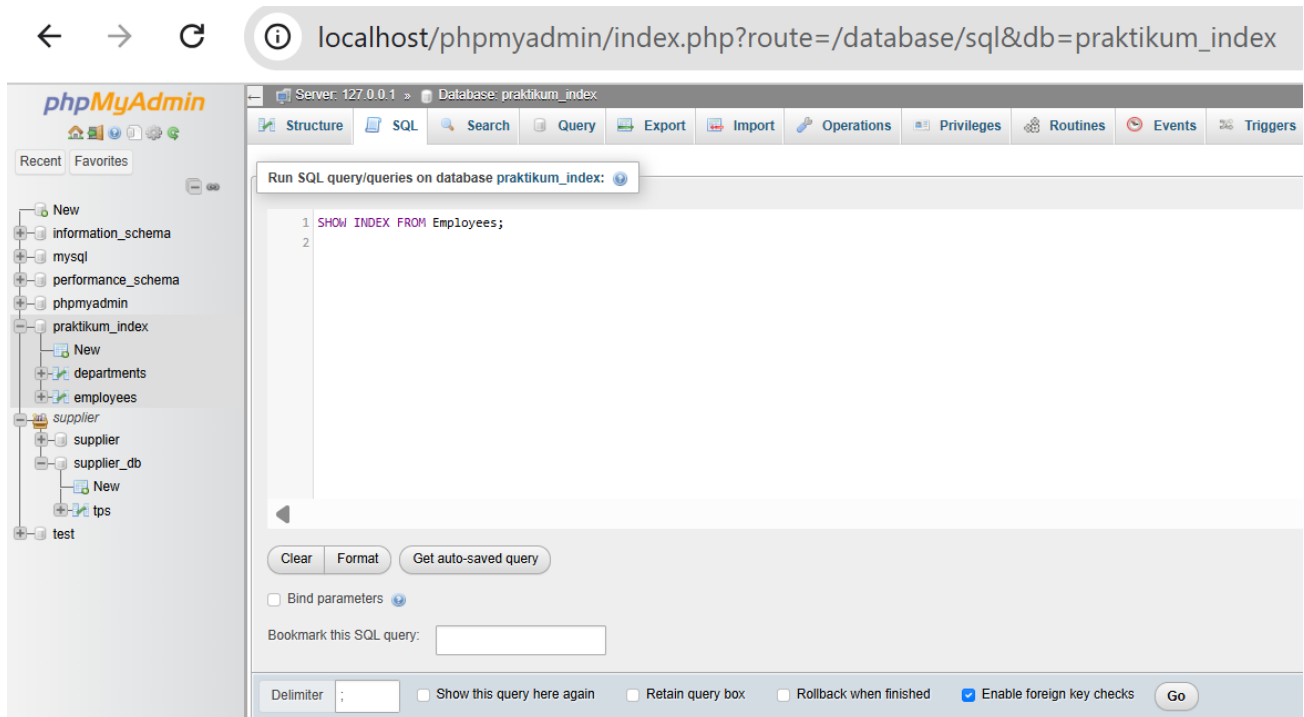
- DROP INDEX adalah perintah untuk menghapus indeks.
- idx_emp_department adalah nama indeks yang akan dihapus.
- Employees adalah tabel tempat indeks berada.



▪ Langkah 2 Verifikasi Penghapusan Indeks:

- Jalankan query berikut untuk memastikan indeks telah dihapus:

SHOW INDEX FROM Employees;

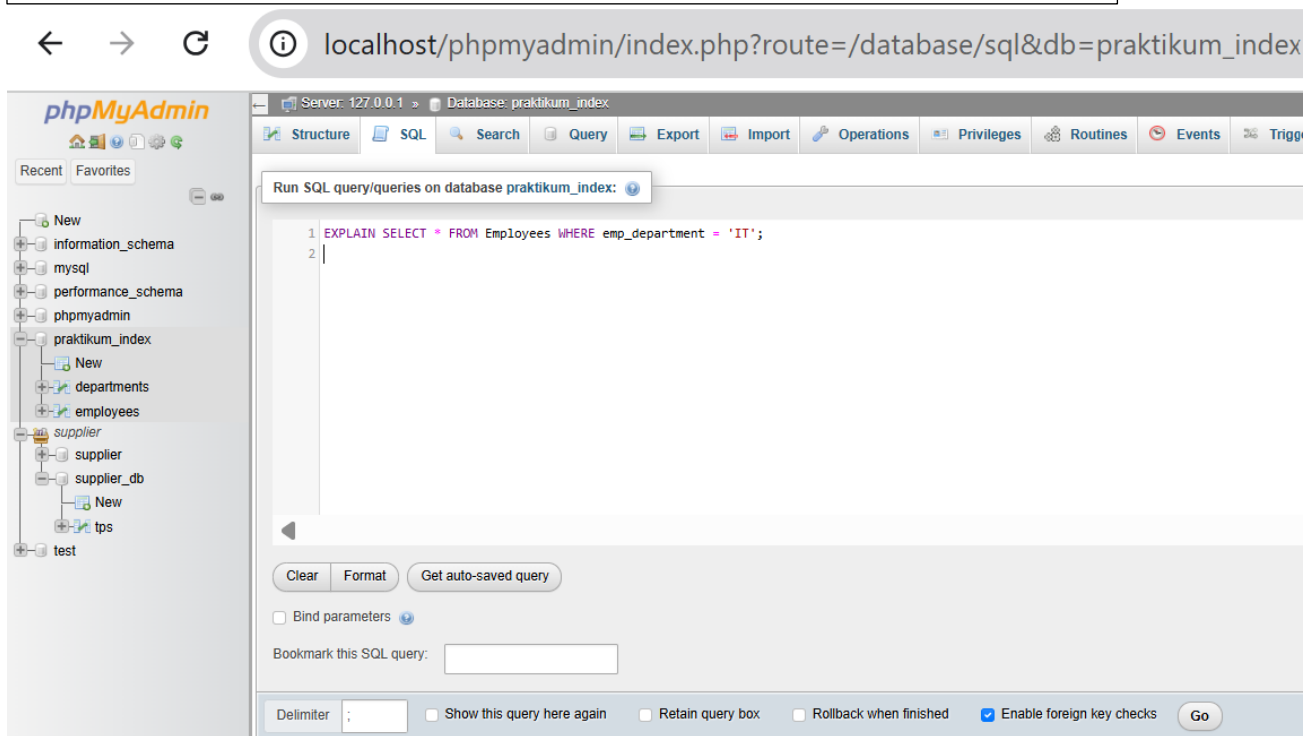


4.5 Analisis dan Tuning Indeks

a. Gunakan EXPLAIN untuk Analisis Query:

- Gunakan perintah EXPLAIN untuk menganalisis bagaimana query dieksekusi dan apakah indeks digunakan:

```
EXPLAIN SELECT * FROM Employees WHERE emp_department = 'IT';
```



Hasil dari EXPLAIN akan menunjukkan:

- Type:** Jenis akses data (misalnya, ALL untuk full table scan, ref untuk penggunaan indeks).
- Possible_keys:** Indeks yang mungkin digunakan.
- Key:** Indeks yang benar-benar digunakan.

- **Rows:** Perkiraan jumlah baris yang dipindai

b. Optimalkan Indeks Berdasarkan Analisis

- **Jika Indeks Tidak Digunakan:**

Jika hasil EXPLAIN menunjukkan bahwa indeks tidak digunakan, mungkin perlu menambahkan indeks baru atau memodifikasi indeks yang ada.

- **Jika Query Masih Lambat:**

Jika query masih lambat meskipun indeks digunakan, pertimbangkan untuk:

- Menambahkan indeks pada kolom lain yang sering digunakan dalam query.
- Menghapus indeks yang tidak diperlukan.
- Mengubah struktur tabel atau query untuk memanfaatkan indeks dengan lebih baik.