

**LAPORAN PRAKTIKUM
BASIS DATA**

**MODUL III
MANIPULASI DAN RETRIEVE DATA**



Disusun Oleh:

Septiandi Nugraha

21104060

SE05-B

**PROGRAM STUDI S1 REKAYASA PERANGKAT LUNAK
FAKULTAS INFORMATIKA
INSTITUT TEKNOLOGI TELKOM PURWOKERTO**

I. Tujuan

1. Dapat menggunakan perintah DML untuk mengelola sebuah data atau baris yang akan diolah
2. Dapat mengisi data pada basis data yang telah dibuat dalam pertemuan sebelumnya

II. Dasar Teori

DML atau Data Manipulation Language merupakan sebuah kelompok perintah yang berfungsi sebagai manipulasi data dalam basis data, semisal untuk pengambilan, penyisipan, pengubahan dan penghapusan data. Perintah yang termasuk dalam kategori DML yaitu:

1. INSERT

Perintah ini digunakan untuk menambahkan baris pada suatu tabel. Terdapat dua cara untuk menambahkan baris yaitu:

- Cara 1 : Menambahkan baris dengan mengisi data pada setiap kolom:

```
INSERT INTO namatabel VALUES (nilai1, nilai2, nilai-n);
```

- Cara 2 : Menambahkan baris dengan hanya mengisi sebuah data pada kolom tertentu:

```
INSERT INTO namatabel (kolom1, kolom2, kolom-n) VALUES (nilai1, nilai2, nilai-n);
```

Keterangan:

Jika sebuah data yang dimasukkan berupa data yang bertipe string, date atau time (contoh: septiandi, rekayasa model perangkat lunak, 2022-03-05) maka pemberian nilainya diapit dengan tanda petik tunggal ('septiandi') atau petik ganda ("gita"). Jika data bertipe numerik (11, 9) maka pemberian nilainya tidak diapit tanda petik tunggal maupun ganda.

2. DELETE

Perintah DELETE digunakan sebagai fungsi untuk menghapus satu baris, baris menggunakan kondisi tertentu atau semua baris. Syntax nya adalah sebagai berikut:

```
DELETE FROM namatabel [WHERE kondisi];
```

Perintah pada tanda [] bersifat opsional untuk menghapus suatu baris menggunakan suatu kondisi tertentu.

3. UPDATE

Perintah UPDATE digunakan untuk mengubah konten data dalam satu atau beberapa kolom dalam sebuah tabel.

```
UPDATE namatabel SET kolom1 = nilai1, kolom2 = nilai2 [WHERE kondisi];
```

Perintah dalam tanda [] bersifat opsional untuk mengubah suatu baris dengan suatu kondisi tertentu.

4. SELECT

Perintah SELECT digunakan untuk menampilkan isi dari suatu tabel yang dapat dikoneksikan dengan tabel lainnya.

- a. Menampilkan sebuah data untuk semua kolom dengan menggunakan simbol asteriks (*):

```
SELECT * FROM namatabel;
```

- b. Menampilkan data untuk beberapa kolom tertentu:

```
SELECT kolom1,kolom2,kolom-n FROM namatabel;
```

- c. Menampilkan sebuah data tertentu dengan klausa WHERE

```
SELECT * FROM namatabel WHERE kondisi;
```

Beberapa operator perbandingan yang dapat digunakan pada klausa WHERE adalah:

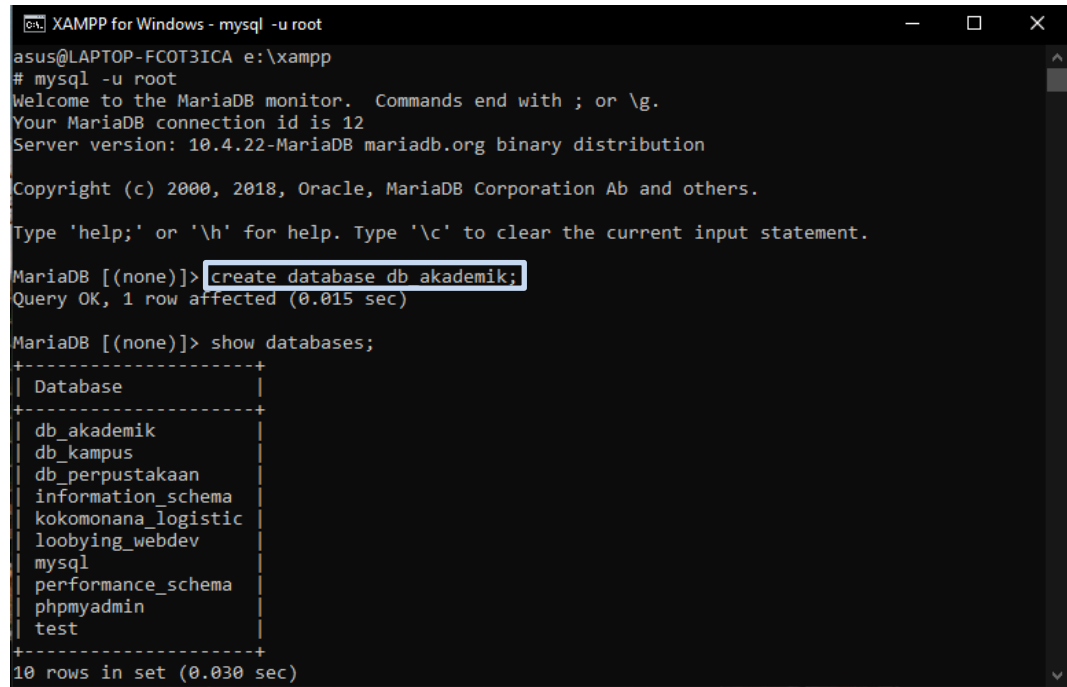
- “=” (sama dengan)
- > (lebih dari)
- < (kurang dari)
- <> (tidak sama dengan)
- >= (lebih dari atau sama dengan)
- <= (kurang dari atau sama dengan)

Adapun operator lain yaitu:

- AND
- OR NOT
- BETWEEN-AND
- IN
- LIKE

III. Praktikum

1. Membuat database dengan nama **db_akademik**



```
XAMPP for Windows - mysql -u root
asus@LAPTOP-FCOT3ICA e:\xampp
# mysql -u root
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 12
Server version: 10.4.22-MariaDB mariadb.org binary distribution

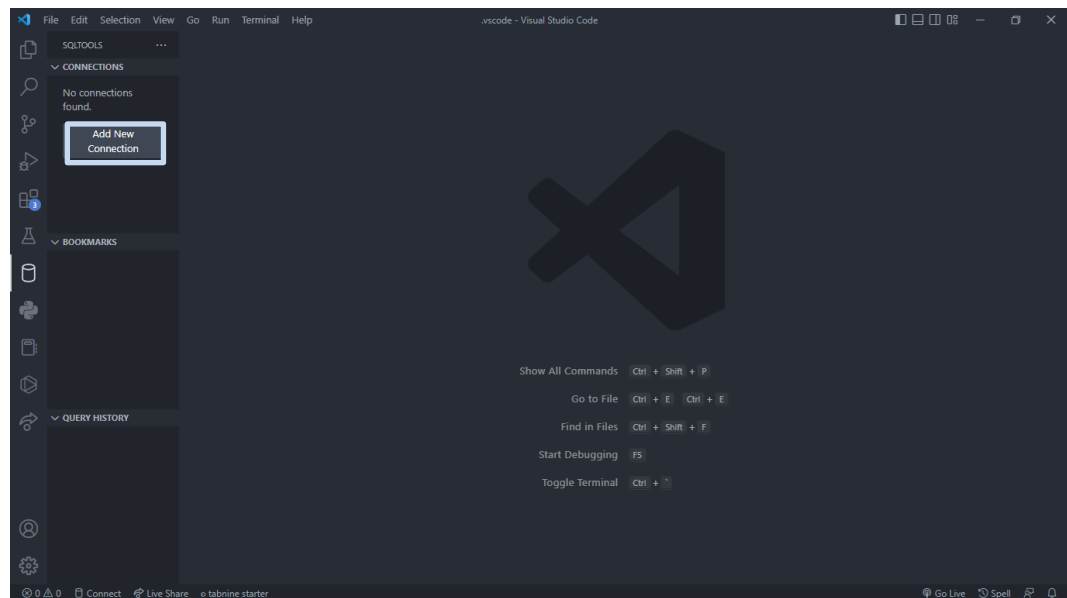
Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

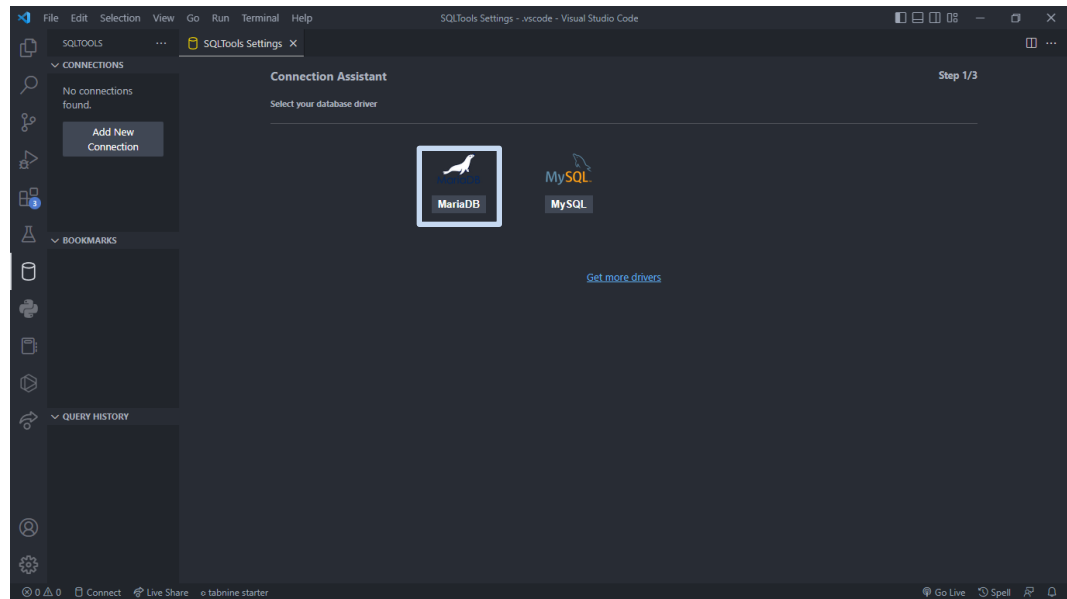
MariaDB [(none)]> create database db akademik;
Query OK, 1 row affected (0.015 sec)

MariaDB [(none)]> show databases;
+-----+
| Database |
+-----+
| db_akademik |
| db_kampus |
| db_perpustakaan |
| information_schema |
| kokomonana_logistic |
| loobying_webdev |
| mysql |
| performance_schema |
| phpmyadmin |
| test |
+-----+
10 rows in set (0.030 sec)
```

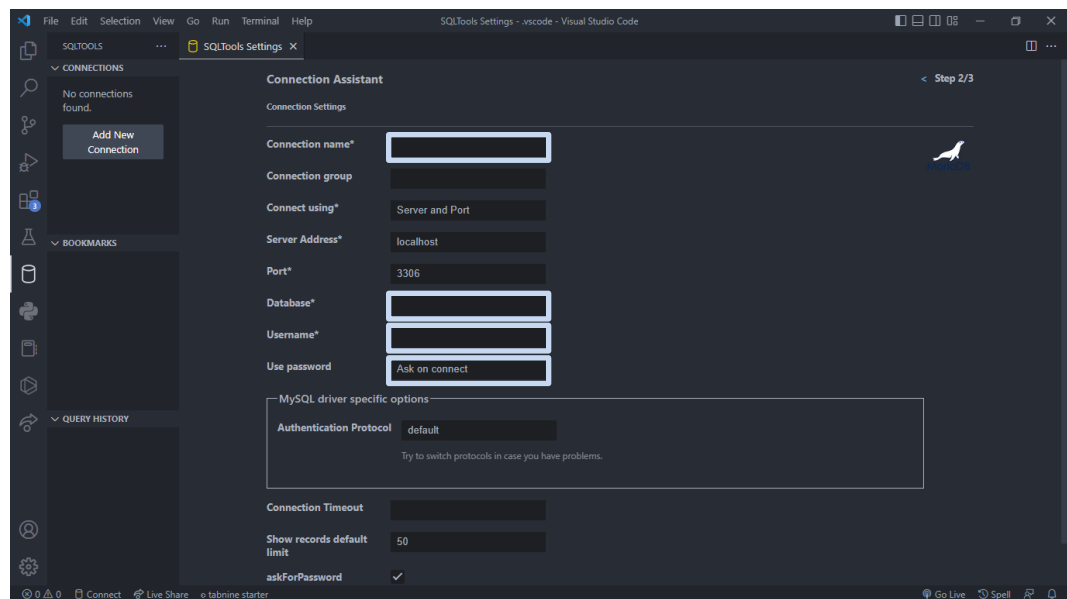
Setelah itu, masuk ke aplikasi Vs Code dan pilih “Add New Connection” untuk menyambungkan database yang sudah dibuat di Shell XAMPP Control Panel



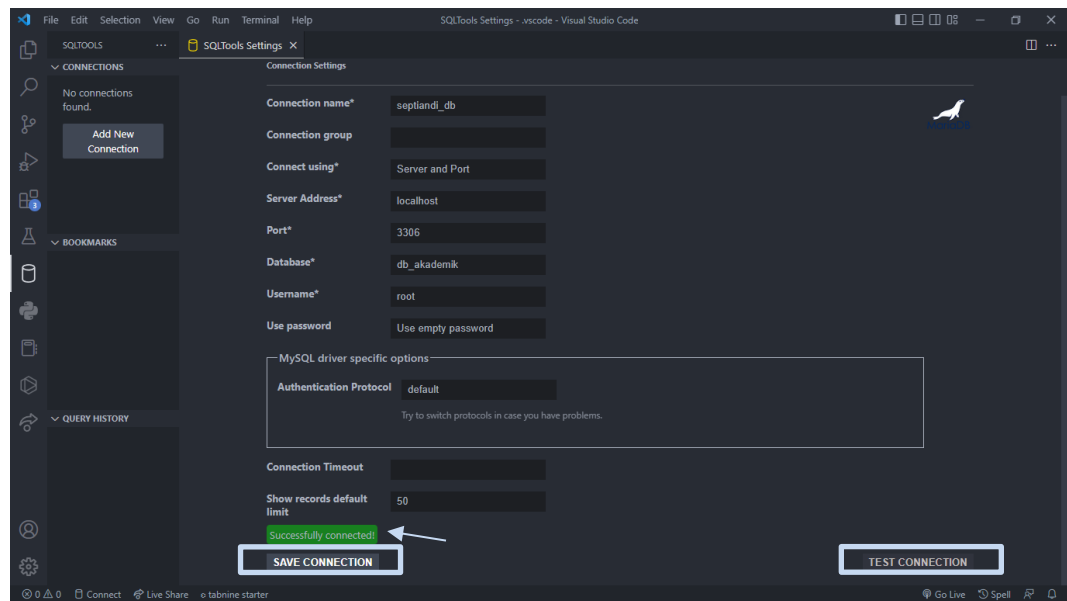
Lalu, pilih MariaDB untuk menyambungkan database



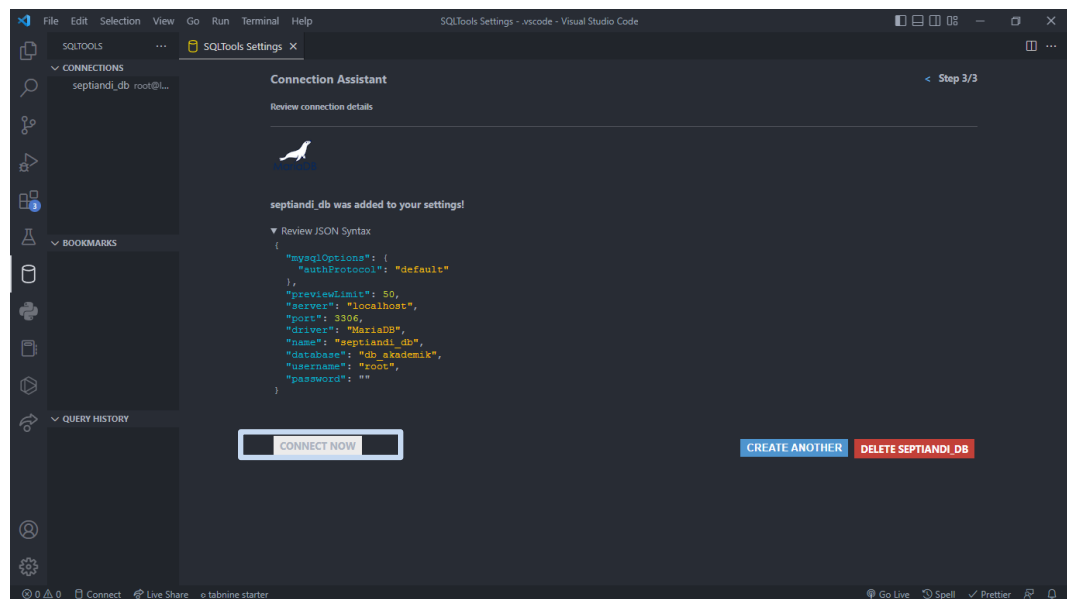
Masukkan form sesuai dengan yang sudah ditentukan



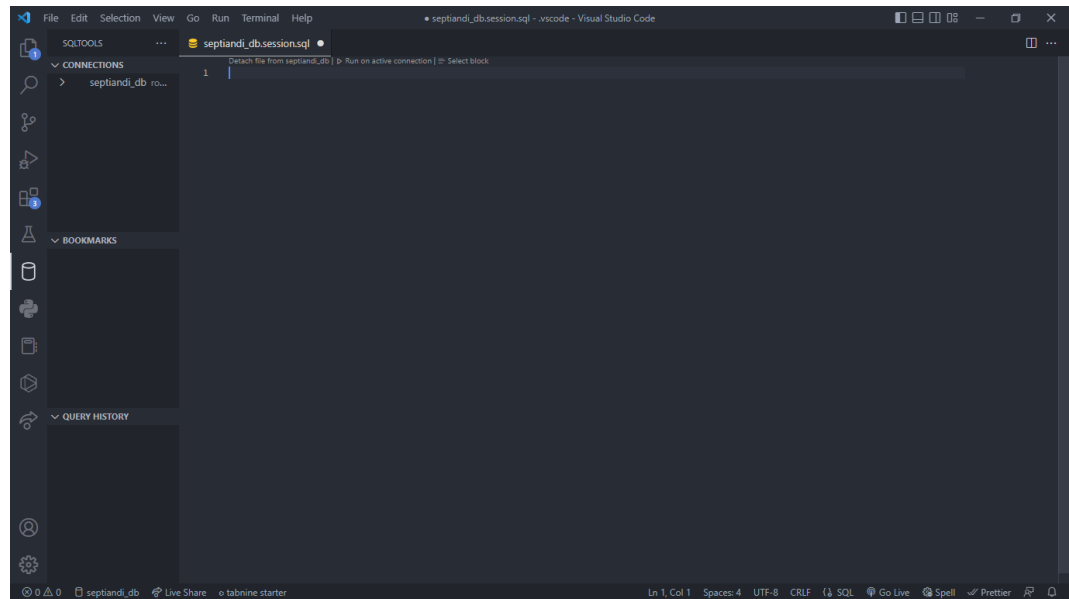
Lalu pilih “Test Connection” untuk mencoba menyambungkan. Ketika sudah muncul pop up yang berwarna hijau dengan teks “Successfully Connected! pilih “Save Connection”



Database pun sudah siap untuk disambungkan, pilih “Connect Now”

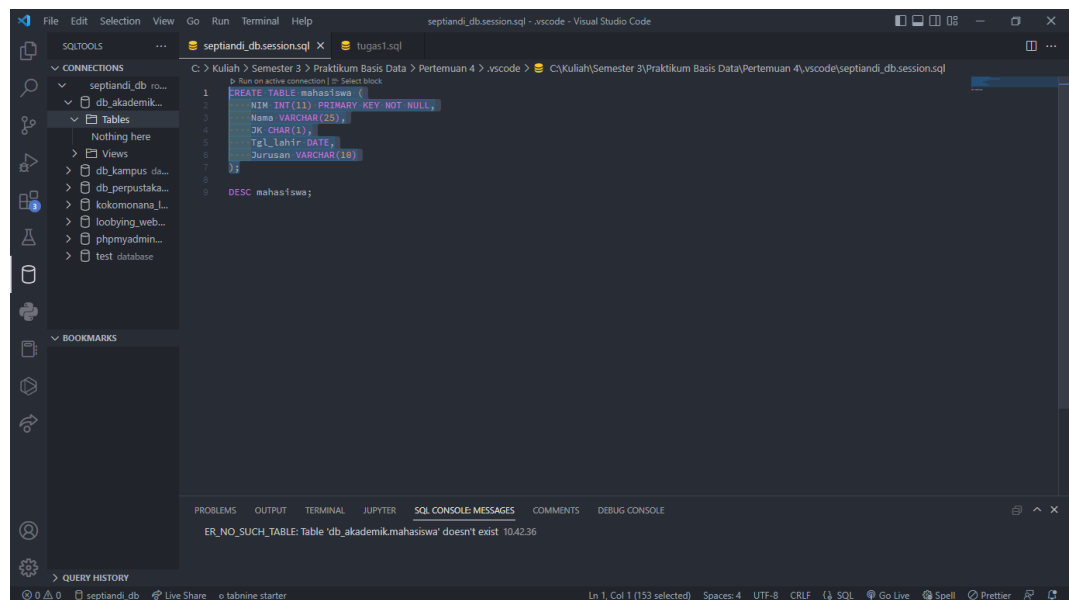


Ketika sudah diarahkan menuju jendela utama, database sudah siap untuk di modifikasi lebih lanjut.

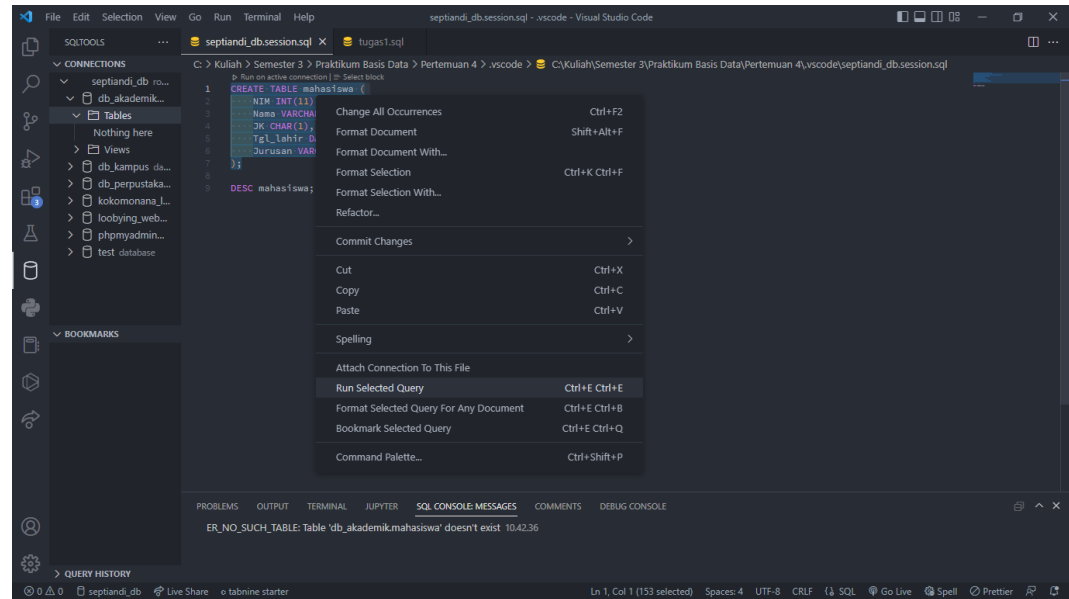


2. Membuat sebuah tabel dengan nama mahasiswa dalam database **db_akademik**

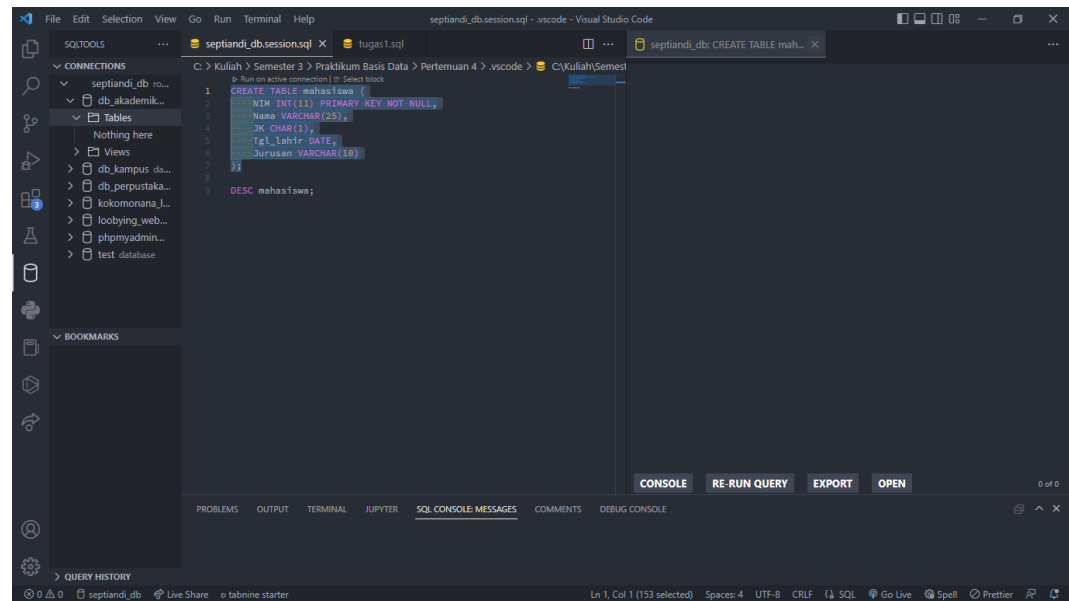
Field	Tipe Data	Type Data
NIM	Int Primary Key	11
Nama	Varchar	25
JK	Char	1
Tgl_lahir	date	
Jurusan	Varchar	10



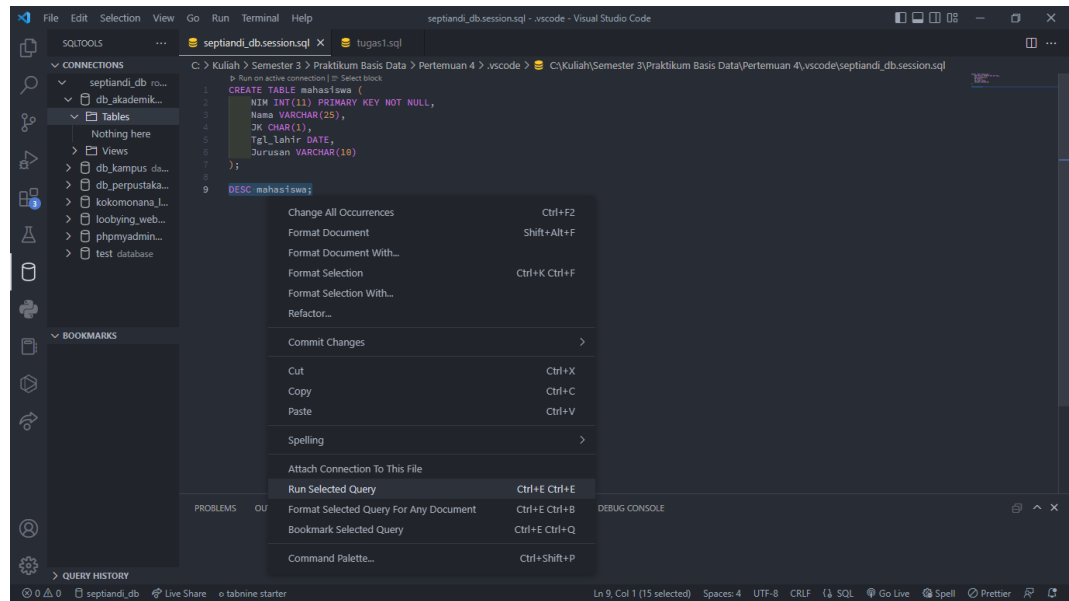
Ketika sudah membuat query nya, pilih tombol kanan di mouse atau touchpad untuk menampilkan menu seperti berikut, lalu pilih “Run Selected Query”



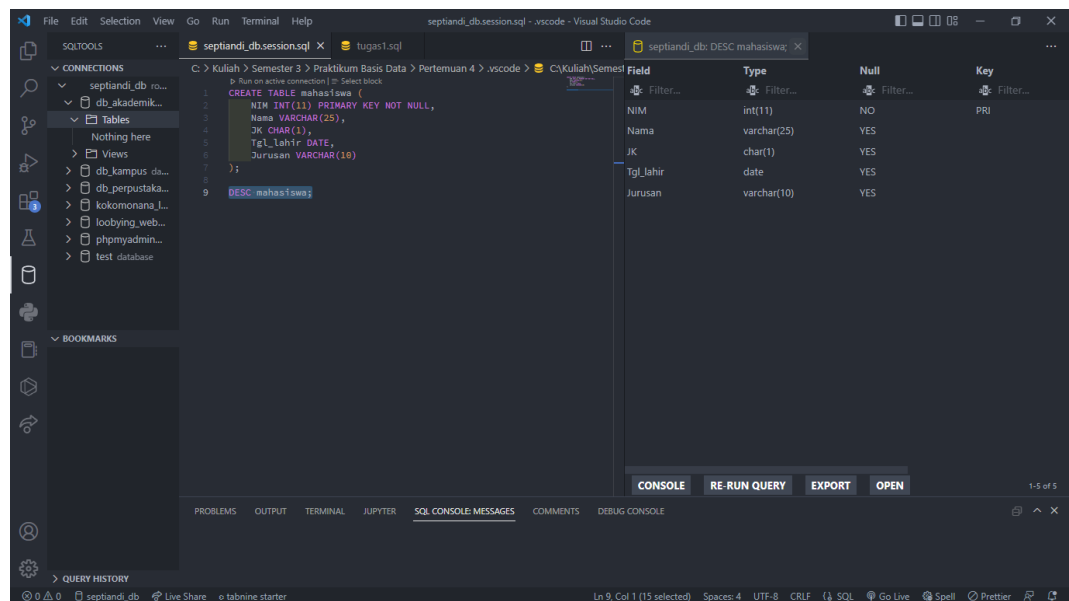
Query sudah tersimpan namun belum bisa ditampilkan



Untuk bisa menampilkan hasil query tipe data yang telah dimasukkan, dapat dilihat sebagai berikut.



Pilih query yang akan dijalankan untuk menampilkan tipe data yang sudah dibuat



Tampilan akan seperti ini.

3. Pengisian data-data berikut ke dalam tabel mahasiswa

NIM	Nama	JK	tgl_lahir	Jurusan
1701001	Maulana	L	1997-02-14	S1 RPL
1701002	Asyifa	P	1998-01-31	S RPL
1701003	Lala	P	1999-07-07	S1 IF
1701004	Hidayat	L	1999-06-20	S1 TI
1701005	Galih	L	1996-02-21	S1 IF
1701006	Zulaika	P	1998-03-20	S1 IF
1701007	Zecko	L	1997-06-05	S1 RPL

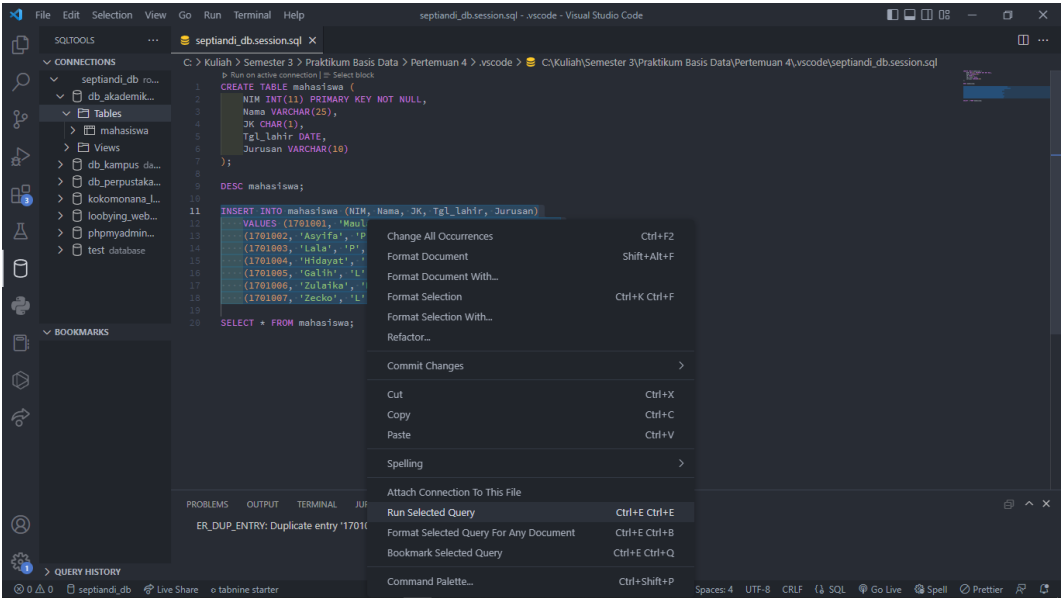
Buat query untuk memasukka isi data, dan pilih teks query yang sudah dibuat

```

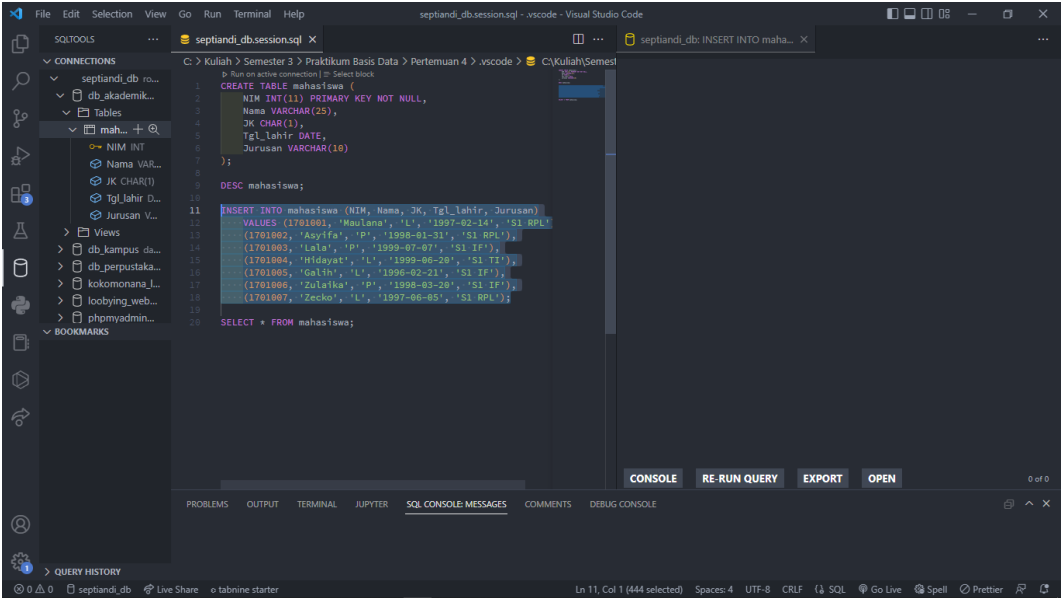
1 CREATE TABLE mahasiswa (
2   NIM INT(11) PRIMARY KEY NOT NULL,
3   Nama VARCHAR(25),
4   JK CHAR(1),
5   tgl_lahir DATE,
6   Jurusan VARCHAR(10)
7 );
8
9 DESC mahasiswa;
10
11 INSERT INTO mahasiswa (NIM, Nama, JK, tgl_lahir, Jurusan)
12 VALUES (1701001, 'Maulana', 'L', '1997-02-14', 'S1 RPL'),
13 (1701002, 'Asyifa', 'P', '1998-01-31', 'S1 RPL'),
14 (1701003, 'Lala', 'P', '1999-07-07', 'S1 IF'),
15 (1701004, 'Hidayat', 'L', '1999-06-20', 'S1 TI'),
16 (1701005, 'Galih', 'L', '1996-02-21', 'S1 IF'),
17 (1701006, 'Zulaika', 'P', '1998-03-20', 'S1 IF'),
18 (1701007, 'Zecko', 'L', '1997-06-05', 'S1 RPL');
19
20 SELECT * FROM mahasiswa;
  
```

ER_DUP_ENTRY: Duplicate entry '1701001' for key 'PRIMARY' 11.08.29

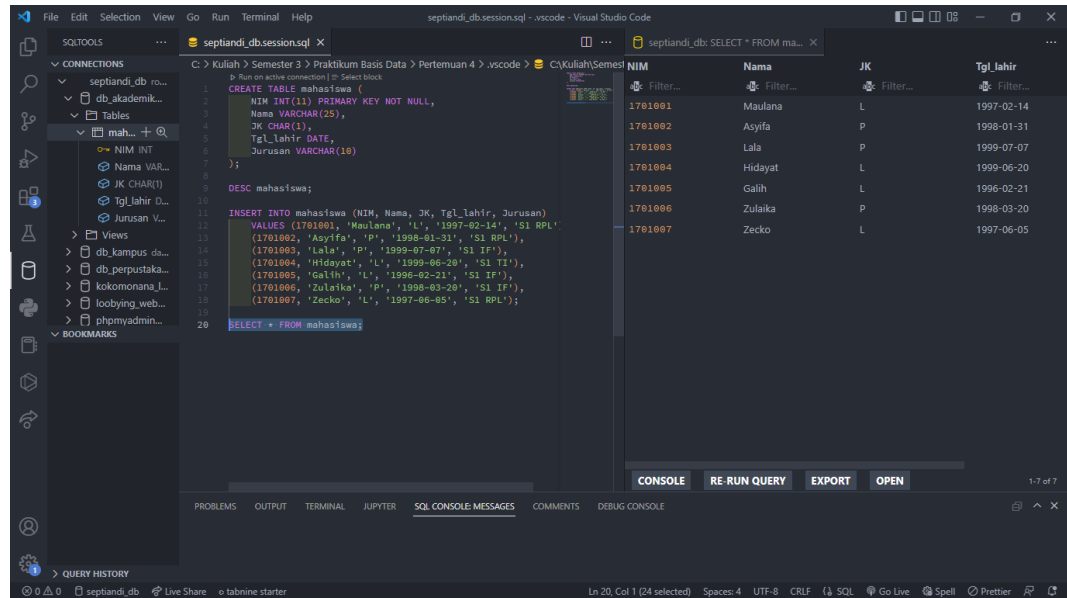
Lalu jalankan query yang sudah dibuat



Query yang sudah dijalankan sudah tersimpan dalam database.



4. Menampilkan semua isi record tabel mahasiswa



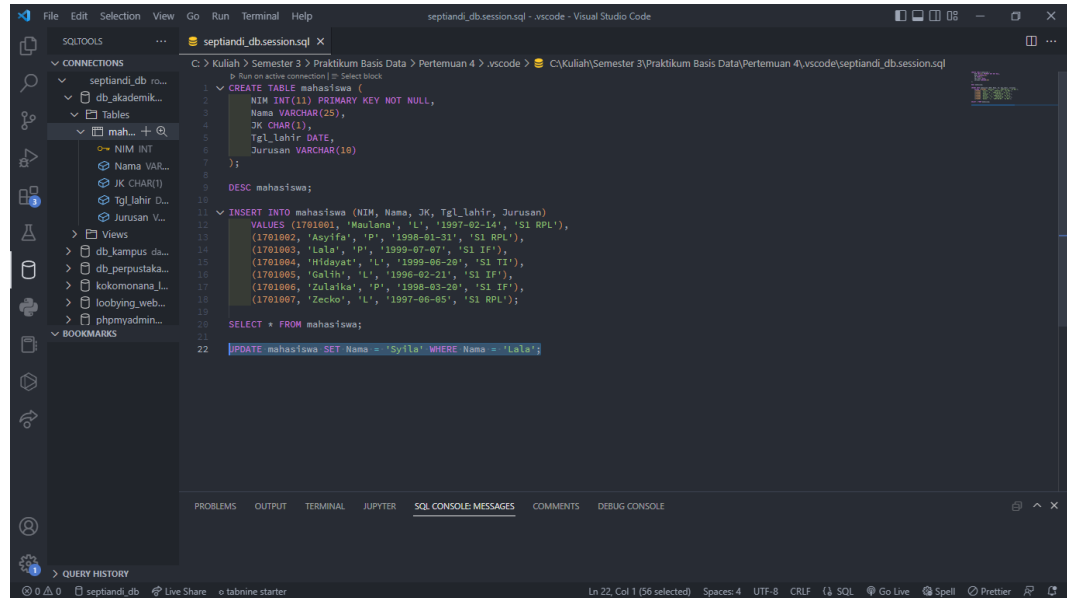
The screenshot shows the Visual Studio Code interface with a SQL file named `septiandi_db.session.sql` open. The code defines a table `mahasiswa` and inserts data into it. The table structure is as follows:

NIM	Nama	JK	Tgl_lahir
1701001	Maulana	L	1997-02-14
1701002	Asyifa	P	1998-01-31
1701003	Lala	P	1999-07-07
1701004	Hidayat	L	1999-06-20
1701005	Galih	L	1996-02-21
1701006	Zulaika	P	1998-03-20
1701007	Zecko	L	1997-06-05

The SQL code in the editor is:

```
1 CREATE TABLE mahasiswa (  
2   NIM INT(11) PRIMARY KEY NOT NULL,  
3   Nama VARCHAR(25),  
4   JK CHAR(1),  
5   Tgl_lahir DATE,  
6   Jurusan VARCHAR(10)  
7 );  
8  
9 DESC mahasiswa;  
10  
11 INSERT INTO mahasiswa (NIM, Nama, JK, Tgl_lahir, Jurusan)  
12 VALUES (1701001, 'Maulana', 'L', '1997-02-14', 'SI RPL'),  
13 (1701002, 'Asyifa', 'P', '1998-01-31', 'SI RPL'),  
14 (1701003, 'Lala', 'P', '1999-07-07', 'SI IF'),  
15 (1701004, 'Hidayat', 'L', '1999-06-20', 'SI TI'),  
16 (1701005, 'Galih', 'L', '1996-02-21', 'SI IF'),  
17 (1701006, 'Zulaika', 'P', '1998-03-20', 'SI IF'),  
18 (1701007, 'Zecko', 'L', '1997-06-05', 'SI RPL');  
19  
20 SELECT * FROM mahasiswa;
```

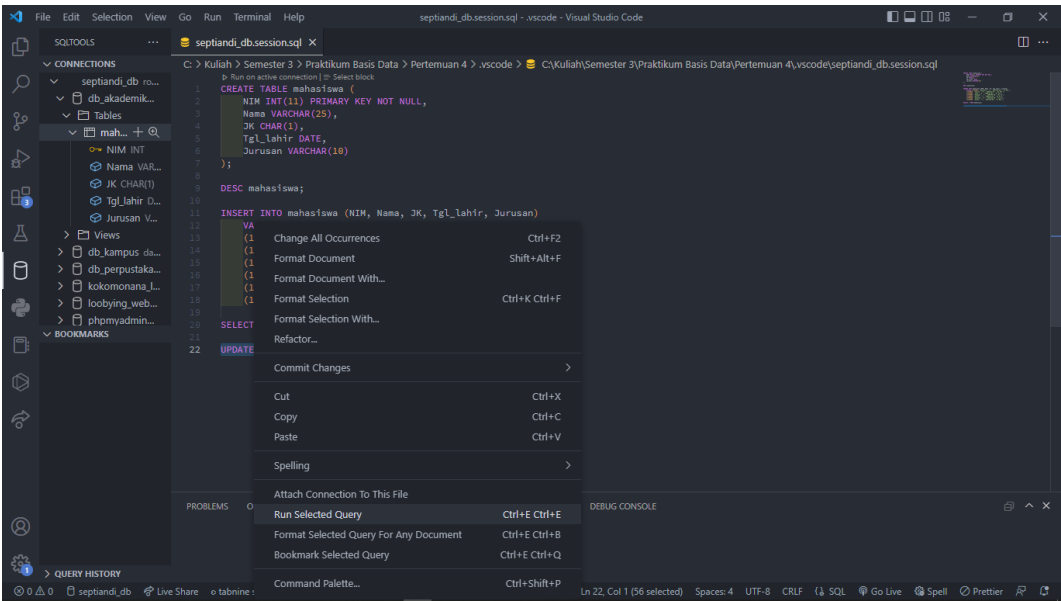
5. Mengubah data salah satu nama mahasiswa yang bernama Lala menjadi Sylla



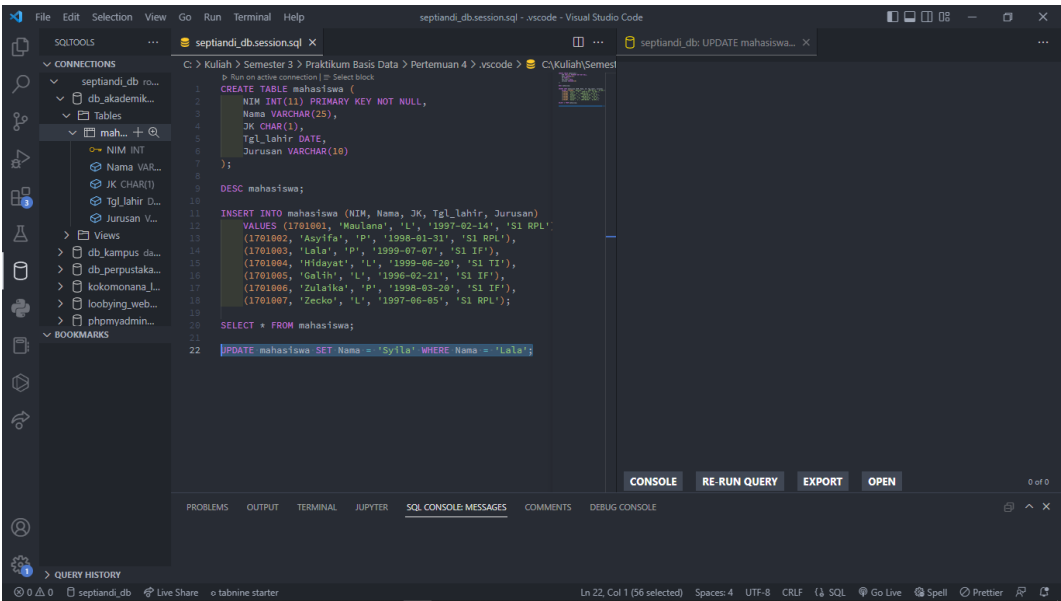
The screenshot shows the Visual Studio Code interface with the same SQL file. The code now includes an `UPDATE` statement to change the name of the student with NIM 1701003 from 'Lala' to 'Sylla'.

```
1 CREATE TABLE mahasiswa (  
2   NIM INT(11) PRIMARY KEY NOT NULL,  
3   Nama VARCHAR(25),  
4   JK CHAR(1),  
5   Tgl_lahir DATE,  
6   Jurusan VARCHAR(10)  
7 );  
8  
9 DESC mahasiswa;  
10  
11 INSERT INTO mahasiswa (NIM, Nama, JK, Tgl_lahir, Jurusan)  
12 VALUES (1701001, 'Maulana', 'L', '1997-02-14', 'SI RPL'),  
13 (1701002, 'Asyifa', 'P', '1998-01-31', 'SI RPL'),  
14 (1701003, 'Lala', 'P', '1999-07-07', 'SI IF'),  
15 (1701004, 'Hidayat', 'L', '1999-06-20', 'SI TI'),  
16 (1701005, 'Galih', 'L', '1996-02-21', 'SI IF'),  
17 (1701006, 'Zulaika', 'P', '1998-03-20', 'SI IF'),  
18 (1701007, 'Zecko', 'L', '1997-06-05', 'SI RPL');  
19  
20 SELECT * FROM mahasiswa;  
21  
22 UPDATE mahasiswa SET Nama = 'Sylla' WHERE Nama = 'Lala';
```

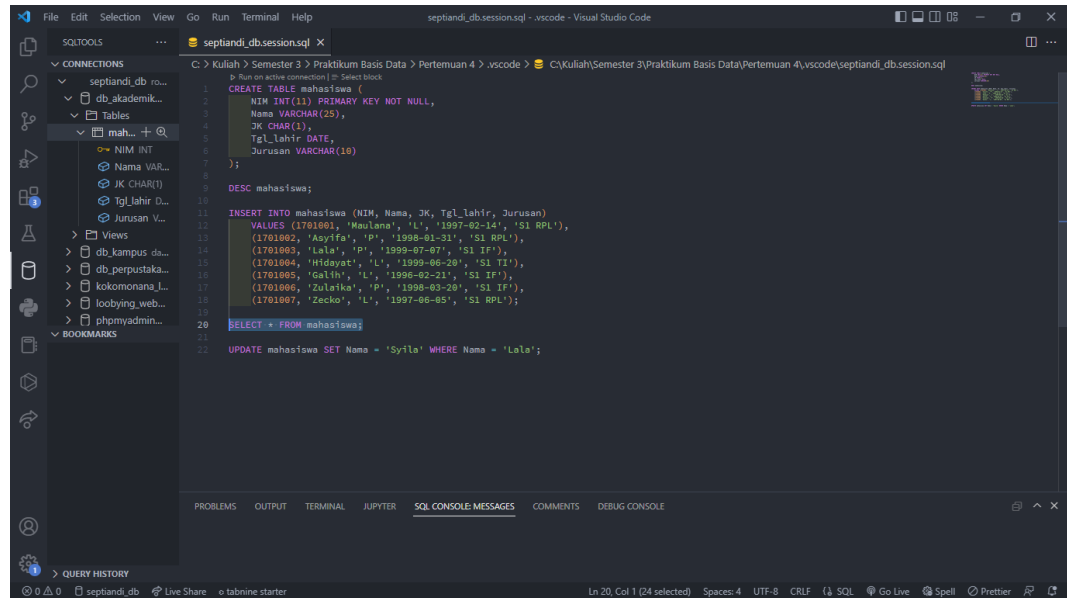
Selanjutnya pilih teks query dan jalankan sesuai dengan pilihan query yang telah ditentukan



Query yang dijalankan sudah tersimpan ke dalam database.

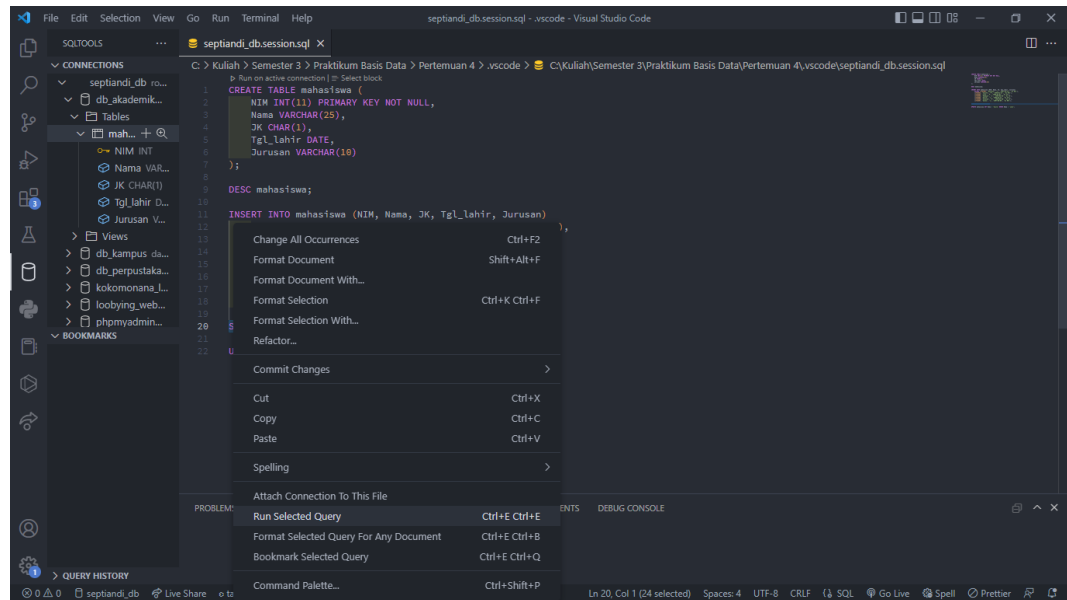


6. Menampilkan satu baris data/record data pada No. 5



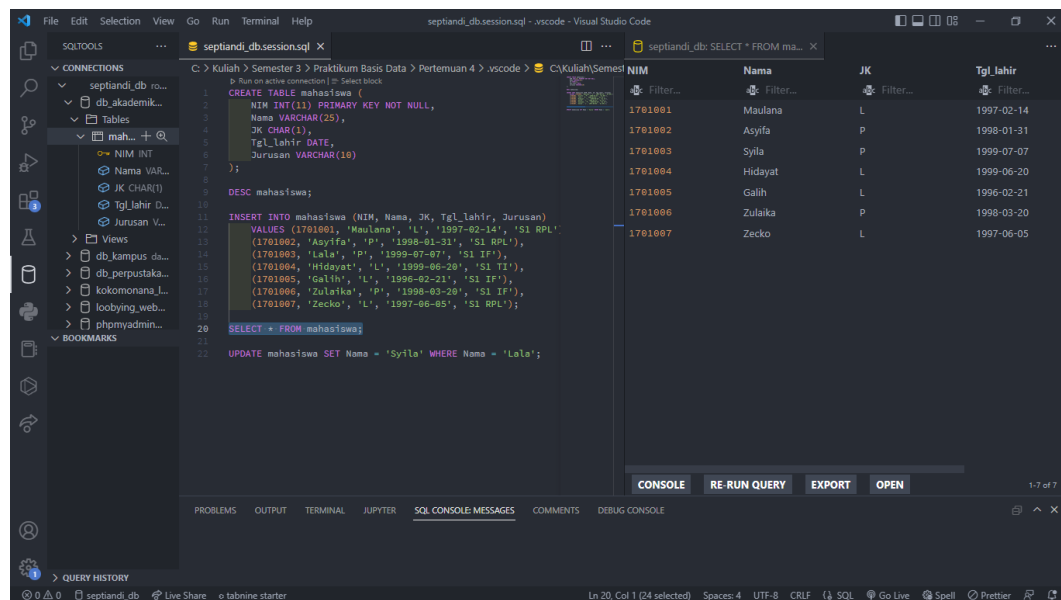
```
1 CREATE TABLE mahasiswa (  
2     NIM INT(11) PRIMARY KEY NOT NULL,  
3     Nama VARCHAR(25),  
4     JK CHAR(1),  
5     Tgl_lahir DATE,  
6     Jurusan VARCHAR(10)  
7 );  
8  
9 DESC mahasiswa;  
10  
11 INSERT INTO mahasiswa (NIM, Nama, JK, Tgl_lahir, Jurusan)  
12 VALUES (1701001, 'Maulana', 'L', '1997-02-14', 'SI RPL'),  
13 (1701002, 'Asyifa', 'P', '1998-01-21', 'SI RPL'),  
14 (1701003, 'Lala', 'P', '1999-07-07', 'SI IF'),  
15 (1701004, 'Hidayat', 'L', '1999-06-20', 'SI TI'),  
16 (1701005, 'Galih', 'L', '1996-02-21', 'SI IF'),  
17 (1701006, 'Zulatika', 'P', '1999-03-20', 'SI IF'),  
18 (1701007, 'Zecko', 'L', '1997-08-03', 'SI RPL');  
19  
20 SELECT * FROM mahasiswa;  
21  
22 UPDATE mahasiswa SET Nama = 'Syila' WHERE Nama = 'Lala';
```

Pilih query yang telah ditulis, lalu jalankan



```
1 CREATE TABLE mahasiswa (  
2     NIM INT(11) PRIMARY KEY NOT NULL,  
3     Nama VARCHAR(25),  
4     JK CHAR(1),  
5     Tgl_lahir DATE,  
6     Jurusan VARCHAR(10)  
7 );  
8  
9 DESC mahasiswa;  
10  
11 INSERT INTO mahasiswa (NIM, Nama, JK, Tgl_lahir, Jurusan)  
12 VALUES (1701001, 'Maulana', 'L', '1997-02-14', 'SI RPL'),  
13 (1701002, 'Asyifa', 'P', '1998-01-21', 'SI RPL'),  
14 (1701003, 'Lala', 'P', '1999-07-07', 'SI IF'),  
15 (1701004, 'Hidayat', 'L', '1999-06-20', 'SI TI'),  
16 (1701005, 'Galih', 'L', '1996-02-21', 'SI IF'),  
17 (1701006, 'Zulatika', 'P', '1999-03-20', 'SI IF'),  
18 (1701007, 'Zecko', 'L', '1997-08-03', 'SI RPL');  
19  
20 SELECT * FROM mahasiswa;  
21  
22 UPDATE mahasiswa SET Nama = 'Syila' WHERE Nama = 'Lala';
```

Data yang sebelumnya sudah dimasukkan, bisa ditampilkan sebagai berikut



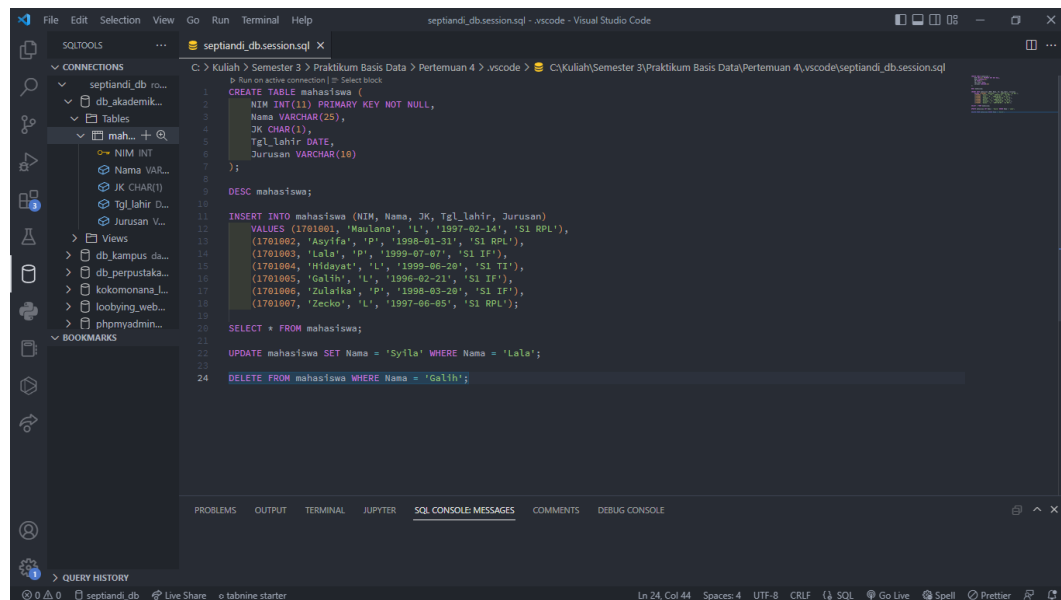
The screenshot shows the Visual Studio Code interface with the SQLTools extension. The left sidebar displays the 'CONNECTIONS' panel with a tree view showing the database structure. The main editor shows a SQL script with the following content:

```
1 CREATE TABLE mahasiswa (  
2     NIM INT(11) PRIMARY KEY NOT NULL,  
3     Nama VARCHAR(25),  
4     JK CHAR(1),  
5     Tgl_lahir DATE,  
6     Jurusan VARCHAR(10)  
7 );  
8  
9 DESC mahasiswa;  
10  
11 INSERT INTO mahasiswa (NIM, Nama, JK, Tgl_lahir, Jurusan)  
12 VALUES (1701001, 'Maulana', 'L', '1997-02-14', 'SI RPL'),  
13 (1701002, 'Asyifa', 'P', '1998-01-31', 'SI RPL'),  
14 (1701003, 'Lala', 'P', '1999-07-07', 'SI IF'),  
15 (1701004, 'Hidayat', 'L', '1999-06-20', 'SI TI'),  
16 (1701005, 'Galih', 'L', '1996-02-21', 'SI IF'),  
17 (1701006, 'Zulaika', 'P', '1998-03-20', 'SI IF'),  
18 (1701007, 'Zecko', 'L', '1997-06-05', 'SI RPL');  
19  
20 SELECT * FROM mahasiswa;  
21  
22 UPDATE mahasiswa SET Nama = 'Syila' WHERE Nama = 'Lala';
```

The right sidebar shows the 'RESULTS' panel with a table of 7 rows of student data:

NIM	Nama	JK	Tgl_lahir
1701001	Maulana	L	1997-02-14
1701002	Asyifa	P	1998-01-31
1701003	Syila	P	1999-07-07
1701004	Hidayat	L	1999-06-20
1701005	Galih	L	1996-02-21
1701006	Zulaika	P	1998-03-20
1701007	Zecko	L	1997-06-05

7. Hapus mahasiswa yang bernama Galih

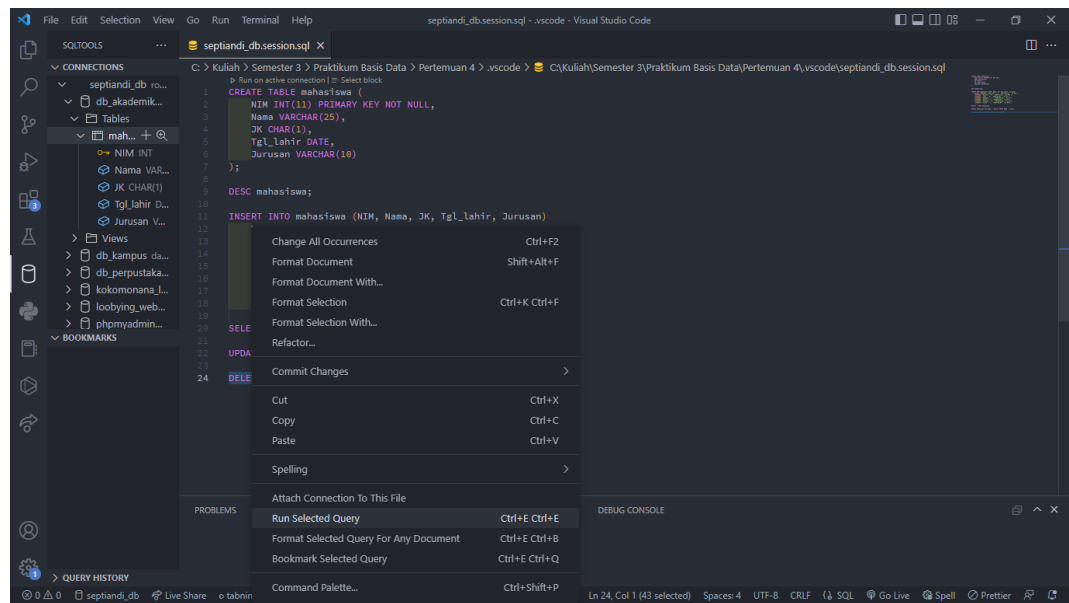


The screenshot shows the Visual Studio Code interface with the SQLTools extension. The left sidebar displays the 'CONNECTIONS' panel with a tree view showing the database structure. The main editor shows a SQL script with the following content:

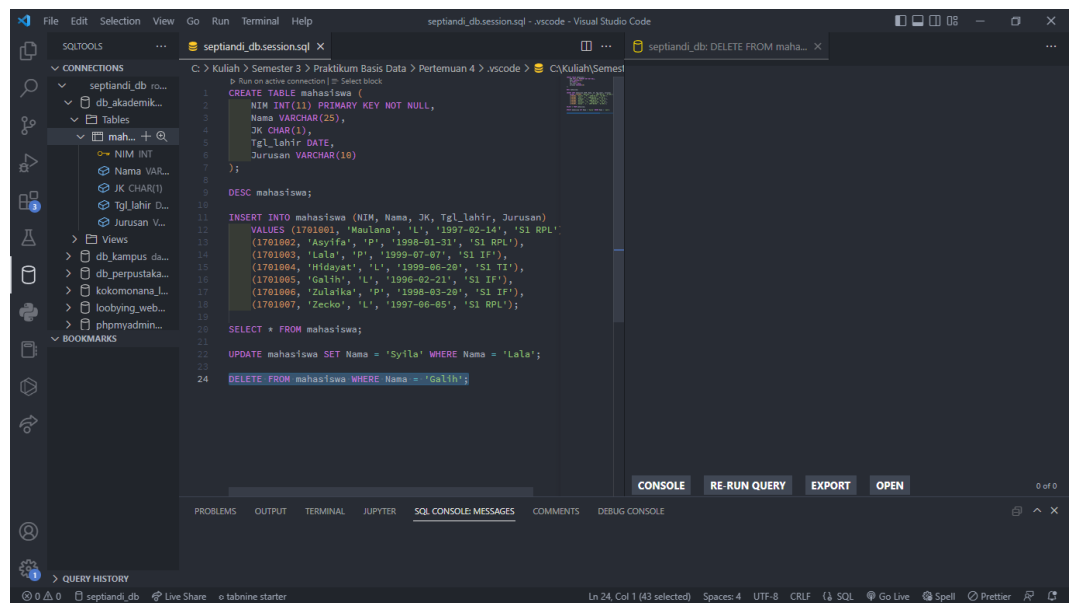
```
1 CREATE TABLE mahasiswa (  
2     NIM INT(11) PRIMARY KEY NOT NULL,  
3     Nama VARCHAR(25),  
4     JK CHAR(1),  
5     Tgl_lahir DATE,  
6     Jurusan VARCHAR(10)  
7 );  
8  
9 DESC mahasiswa;  
10  
11 INSERT INTO mahasiswa (NIM, Nama, JK, Tgl_lahir, Jurusan)  
12 VALUES (1701001, 'Maulana', 'L', '1997-02-14', 'SI RPL'),  
13 (1701002, 'Asyifa', 'P', '1998-01-31', 'SI RPL'),  
14 (1701003, 'Lala', 'P', '1999-07-07', 'SI IF'),  
15 (1701004, 'Hidayat', 'L', '1999-06-20', 'SI TI'),  
16 (1701005, 'Galih', 'L', '1996-02-21', 'SI IF'),  
17 (1701006, 'Zulaika', 'P', '1998-03-20', 'SI IF'),  
18 (1701007, 'Zecko', 'L', '1997-06-05', 'SI RPL');  
19  
20 SELECT * FROM mahasiswa;  
21  
22 UPDATE mahasiswa SET Nama = 'Syila' WHERE Nama = 'Lala';  
23  
24 DELETE FROM mahasiswa WHERE Nama = 'Galih';
```

The right sidebar shows the 'RESULTS' panel with a message indicating the deletion of 1 row:

```
1 row(s) deleted.
```

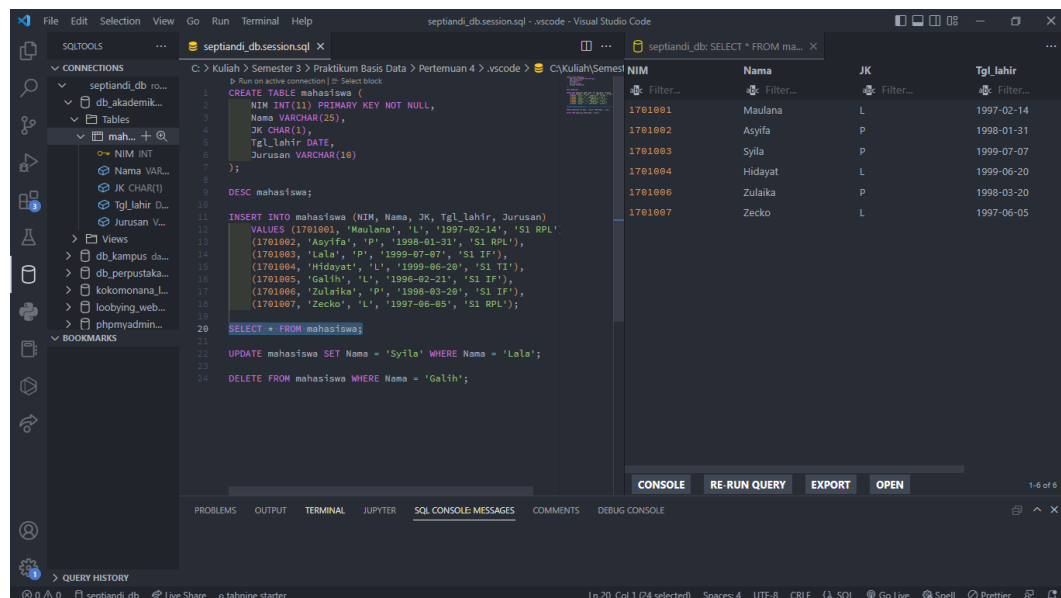


Pilih query lalu jalankan



Query sudah disimpan dalam database

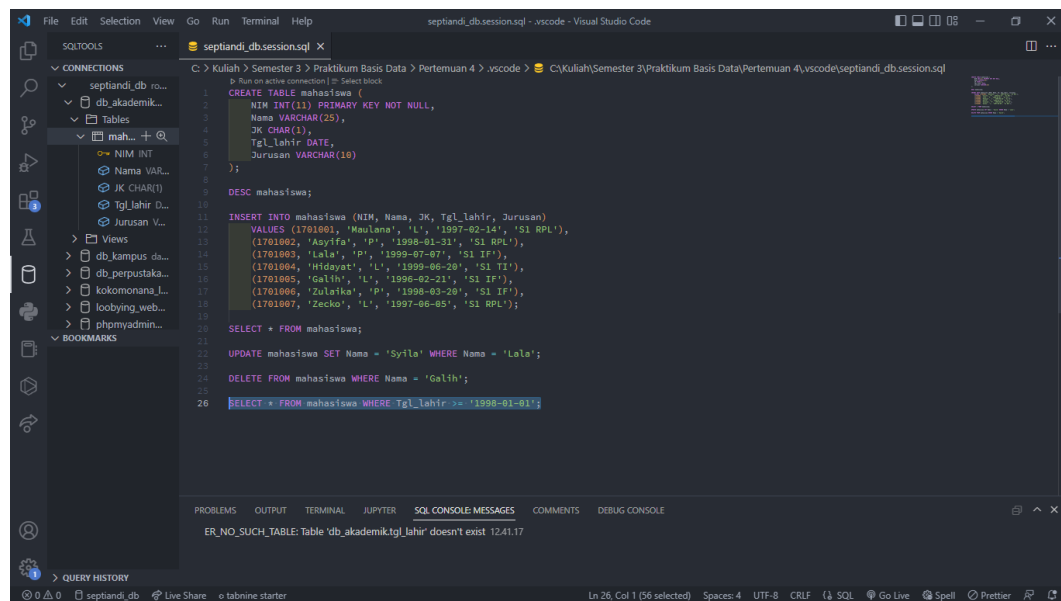
Data yang bernama “Galih” sudah terhapus dari tabel mahasiswa.



The screenshot shows the Visual Studio Code interface with a SQL file named `septiandi_db.session.sql` open. The code defines a table `mahasiswa` with columns `NIM`, `Nama`, `JK`, and `Tgl_lahir`. It includes `INSERT` statements for several students, including Galih. The `SELECT * FROM mahasiswa;` query is highlighted. On the right, a table view displays the data:

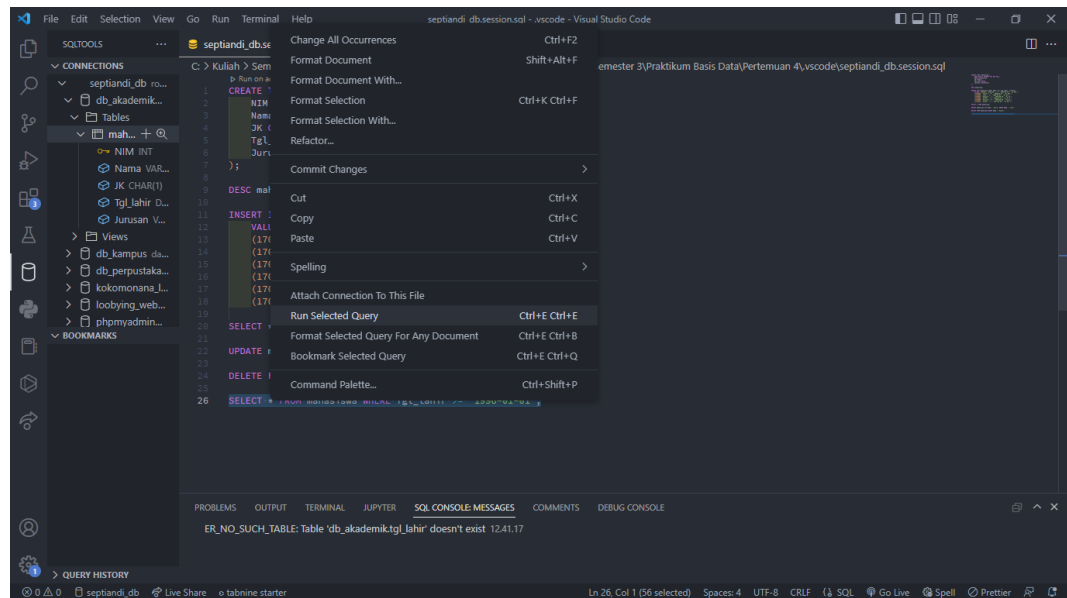
NIM	Nama	JK	Tgl_lahir
1701001	Maulana	L	1997-02-14
1701002	Asyifa	P	1998-01-31
1701003	Syila	P	1999-07-07
1701004	Hidayat	L	1999-06-20
1701006	Zulaika	P	1998-03-20
1701007	Zecko	L	1997-06-05

8. Tampilkan data/record yang tanggal kelahirannya lebih dari atau sama dengan 1998-01-01

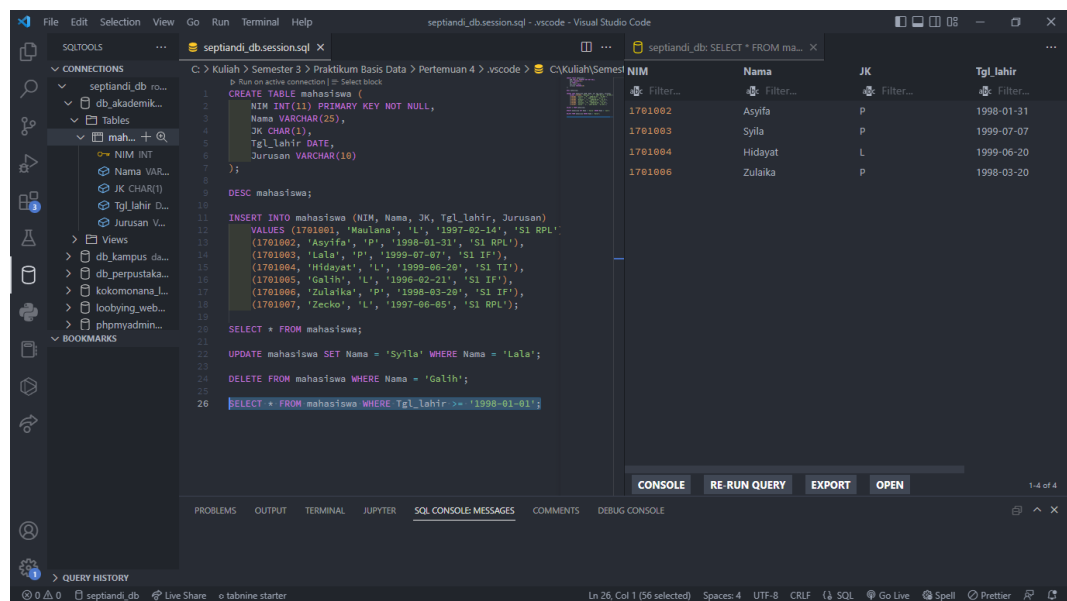


The screenshot shows the Visual Studio Code interface with the same SQL file. The query `SELECT * FROM mahasiswa WHERE Tgl_lahir >= '1998-01-01';` is highlighted. The bottom panel shows an error message: `ER_NO_SUCH_TABLE: Table 'db_akademik.tgl_lahir' doesn't exist 12.41.17`.

Pilih query yang sudah ditulis, lalu jalankan



Hasil tampilan akan seperti ini, perintah query akan menentukan data yang ditampilkan sesuai dengan perintah tanggal lahir yang sama atau lebih dari tanggal lahir yang telah ditentukan.



IV. Kesimpulan

Dalam membuat sebuah database di VS Code akan lebih mudah, dikarenakan dalam Vs Code sudah tertera beberapa fitur yang memudahkan untuk keperluan programming dan membuat database. Jika membuat database dalam Shell XAMPP Control Panel, ketika ada kesalahan penulisan dalam query, akan error dan ditulis

ulang. Tetapi dalam VS Code, kesalahan penulisan dalam menulis query bisa langsung dibenarkan baris per baris dan lebih menghemat waktu.