

LAPORAN AKHIR PRAKTIKUM

Mata Praktikum : Sistem Basis Data 1
Kelas : 3IA24
Praktikum ke- : 1
Tanggal : 21 November 2023
Materi : Pengenalan Basis Data & DML
NPM : 50421704
Nama : Juan Samuel Christopher
Ketua Asisten : Madanil
Nama Asisten :
Paraf Asisten :
Jumlah Lembar : 9



LABORATORIUM TEKNIK INFORMATIKA

UNIVERSITAS GUNADARMA

2023

LISTING

```
Query Query History
1 CREATE TABLE Pelanggan (
2     id_pelanggan VARCHAR (20) NOT NULL PRIMARY KEY,
3     nama_pelanggan VARCHAR (30),
4     umur INT,
5     alamat VARCHAR (50),
6     kota VARCHAR (30)
7 );
8
9 SELECT * FROM Pelanggan;
```

```
juan_mart/postgres@PostgreSQL 15
Query Query History
1 ALTER TABLE Pelanggan DROP COLUMN kota;
2
3 SELECT * FROM Pelanggan;
```

```
juan_mart/postgres@PostgreSQL 15
Query Query History
1 INSERT INTO Pelanggan (id_pelanggan, nama_pelanggan, umur, alamat)
2 VALUES
3 ('01', 'Juan', 20, 'Perumahan Ceunah'),
4 ('02', 'Wirnu', 20, 'Perumahan NutriSari'),
5 ('03', 'Marcelino', 20, 'Perumahan Fortinaiti'),
6 ('04', 'Hasyimadminshope', 21, 'Perumahan Oren');
7
8 SELECT * FROM Pelanggan;
```

```
juan_mart/postgres@PostgreSQL 15
Query Query History
1 UPDATE Pelanggan SET nama_pelanggan = 'Hasyim'
2 WHERE id_pelanggan = '04';
3
4 SELECT * FROM Pelanggan;
```

```
juan_mart/postgres@PostgreSQL 15
Query Query History
1 DELETE FROM Pelanggan WHERE id_pelanggan = '02';
2
3 SELECT * FROM Pelanggan;
```

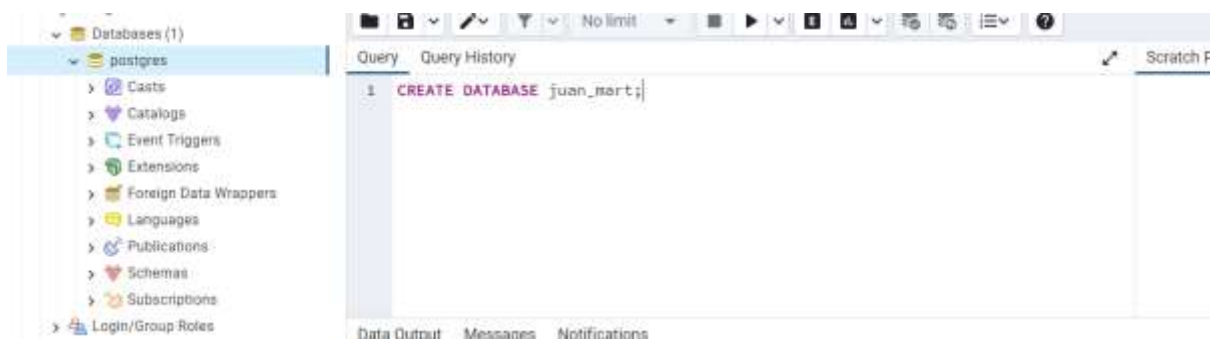
LOGIKA

Pada pertemuan 1 ini , disini kita belajar cara membuat sebuah database yang berisikan table pada program aplikasi postgres. Sistem basis data adalah suatu sistem yang terdiri dari kumpulan data yang saling berhubungan secara logis, dan deskripsi dari data-data tersebut serta dirancang untuk memenuhi informasi yang dibutuhkan oleh sebuah organisasi. Sistem basis data terdiri dari dua komponen utama, yaitu basis data dan sistem manajemen basis data (DBMS).

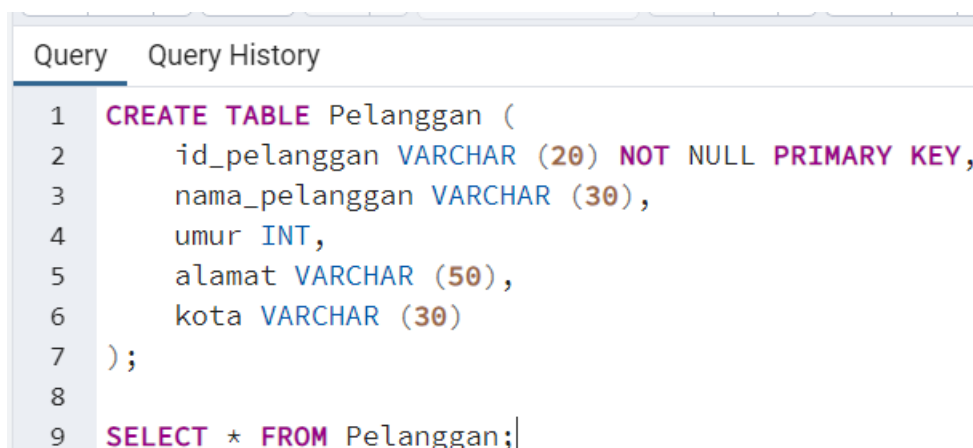
Basis data adalah kumpulan data yang terorganisir dan dihubungkan secara logis. Data dalam basis data dapat berupa teks, angka, gambar, atau video. Basis data dirancang untuk memenuhi kebutuhan informasi dari penggunaanya.

Sistem manajemen basis data (DBMS) adalah perangkat lunak yang digunakan untuk mengelola basis data. DBMS menyediakan berbagai fungsi untuk mengelola basis data, seperti: Menyimpan data, Mengambil data, Mengubah data, Menghapus data, Memproteksi data

Berikut Langkah-langkah dan logika pengerjaan Activity pada pertemuan 1 minggu ini :

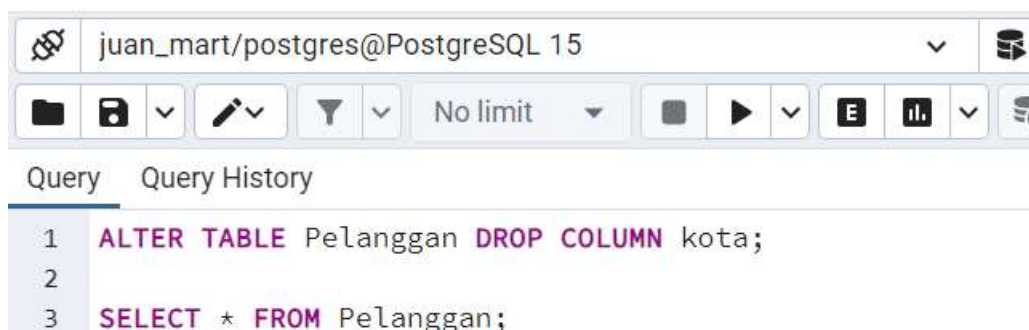


Langkah pertama yang saya lakukan adalah membuat database dengan sintaks **CREATE DATABASE** juan_mart; dimana program akan membuat database yang bernama **juan_mart**. Kemudian refresh Databasenya maka database baru bernama **juan_mart** akan muncul. Klik kiri pada database juan_mart lalu klik **query tool** yang akan membawa kita ke field kosong untuk mengetik perintah SQL.

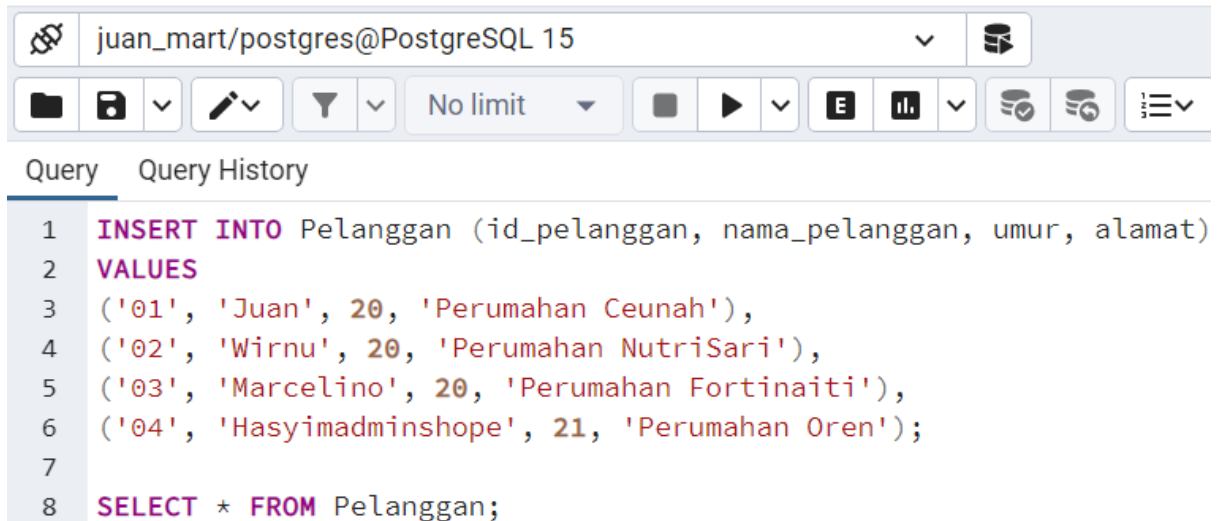


Selanjutnya saya membuat table baru yang bernama **Pelanggan** dengan sintak **CREATE TABLE** Disini saya memasukkan 5 kolom data yaitu **id_pelanggan VARCHAR (20) NOT NULL PRIMARY KEY**, yang bertujuan untuk membuat kolom bernama **id_pelanggan** memiliki tipe data **VARCHAR** (tipe data string yang dapat menampung data teks dengan panjang bervariasi, max 255 karakter yang akan disimpan) yang memiliki panjang data 20 karakter, **NOT NULL** berarti kolom ini tidak boleh kosong / harus mempunyai isi, lalu **PRIMARY KEY** adalah untuk mengidentifikasi setiap baris atau record secara unik dalam sebuah tabel database.

Kemudian **nama_pelanggan VARCHAR (30)** bertujuan untuk membuat kolom baru bernama **nama_pelanggan** bertipe data **VARCHAR** dengan Panjang data 30 karakter, lalu **umur INT** bertujuan untuk membuat kolom baru bernama **umur** bertipe data **INT** tipe data numerik integer yang digunakan untuk menyimpan bilangan bulat. Kemudian **alamat VARCHAR (50)** bertujuan untuk membuat kolom baru bernama **alamat** bertipe data **VARCHAR** dengan Panjang data 50 karakter. Kemudian **kota VARCHAR (30)** bertujuan untuk membuat kolom baru bernama **kota** bertipe data **VARCHAR** dengan Panjang data 30 karakter. Lalu **SELECT * FROM Pelanggan;** yang akan mengambil data dari table **Pelanggan** dan menampilkan nya pada field output.

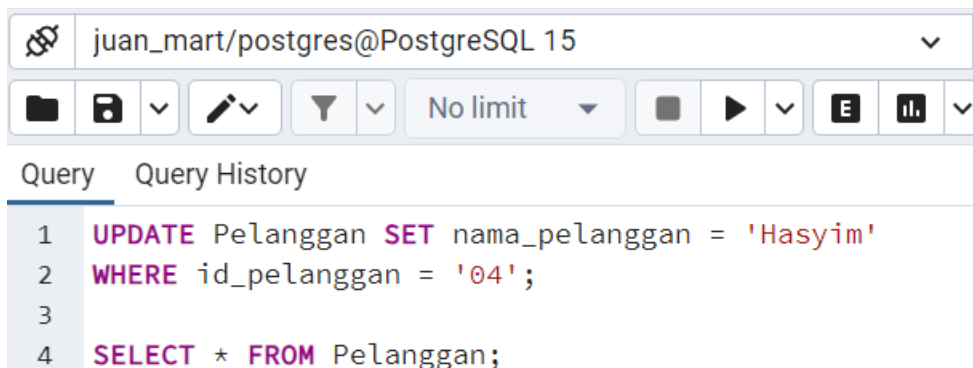


Langkah selanjutnya saya akan menghapus kolom **kota** menggunakan sintaks seperti ini **ALTER TABLE Pelanggan DROP COLUMN kota;** yang dimana **ALTER TABLE Pelanggan** digunakan untuk mengubah struktur dalam database **Pelanggan** dan **DROP COLUMN kota** yang berarti **DROP** berfungsi untuk menghapus kolom **kota** dari table.



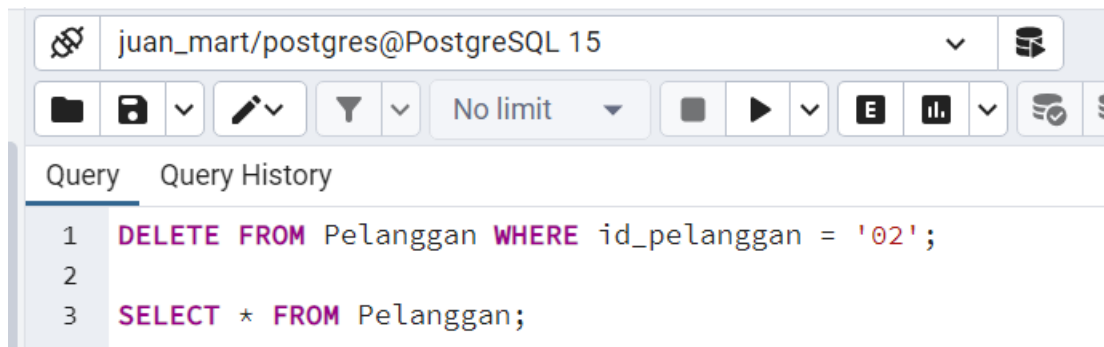
```
1 INSERT INTO Pelanggan (id_pelanggan, nama_pelanggan, umur, alamat)
2 VALUES
3 ('01', 'Juan', 20, 'Perumahan Ceunah'),
4 ('02', 'Wirnu', 20, 'Perumahan NutriSari'),
5 ('03', 'Marcelino', 20, 'Perumahan Fortinaiti'),
6 ('04', 'Hasyimadminshope', 21, 'Perumahan Oren');
7
8 SELECT * FROM Pelanggan;
```

Langkah selanjutnya saya akan menambahkan 4 records ke dalam table **Pelanggan** menggunakan **INSERT INTO** lalu spesifikasikan nama nama kolom apa saja dan values (isi/nilai) dari kolom yang akan diisi. Seperti diatas maka '01' adalah **id_pelanggan**, 'Juan' adalah **nama_pelanggan**, 20 adalah **umur**, 'Perumahan Ceunah' adalah **alamat**. Jadi masing masing kolom sudah memiliki 1 buah record. Langkah untuk menambahkan 3 record selanjutnya sama seperti tadi.



```
1 UPDATE Pelanggan SET nama_pelanggan = 'Hasyim'
2 WHERE id_pelanggan = '04';
3
4 SELECT * FROM Pelanggan;
```

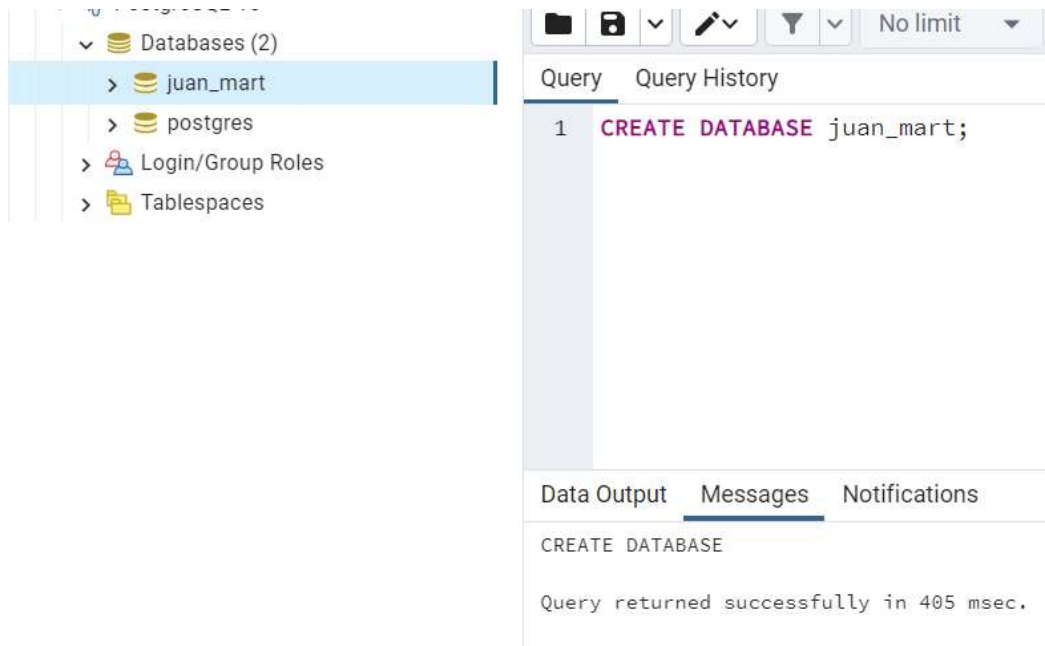
Langkah selanjutnya saya akan mengupdate salah satu data pelanggan , menggunakan **UPDATE Pelanggan** yang berfungsi untuk memodifikasi records yang ada di table **Pelanggan**, lalu **SET nama_pelanggan** akan memberi nilai record baru untuk kolom **nama_pelanggan** yaitu 'Hasyim', **WHERE id_pelanggan = '04'** adalah dimana lokasi data yang ingin diubah disini **id_pelanggan = '04'** sebagai primary key untuk data ke 4. Maka pada id_pelanggan ke 4 , record di dalam kolom nama_pelanggannya akan dimodifikasi menjadi 'Hasyim"



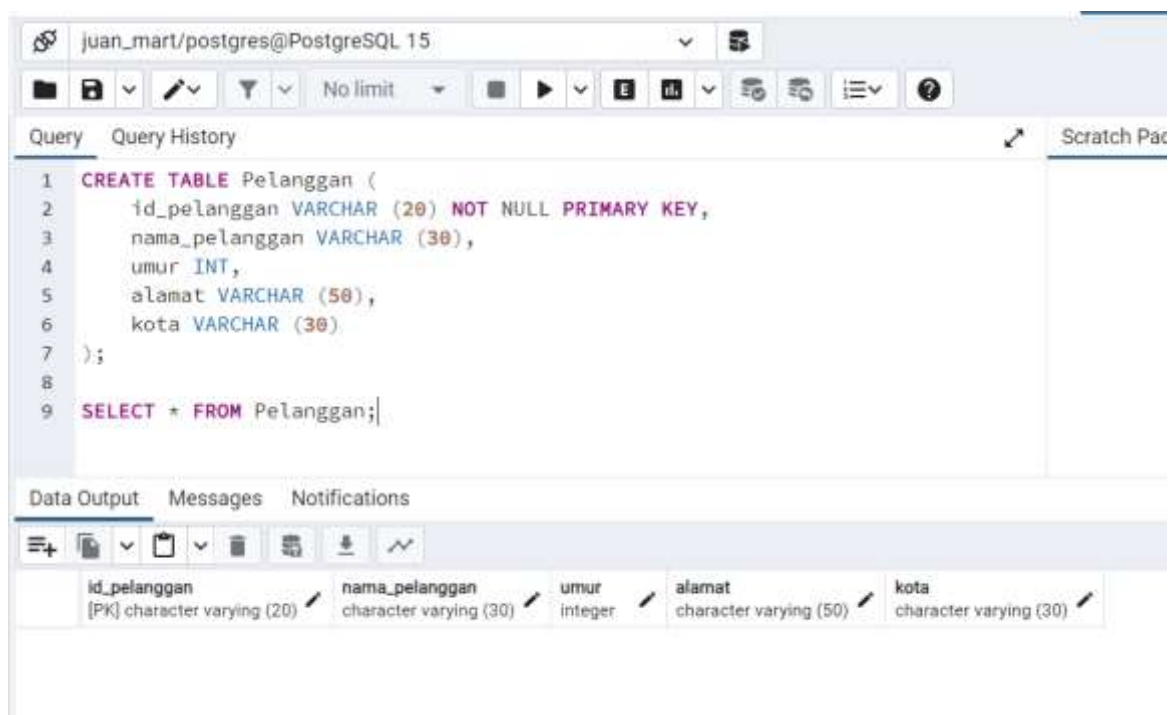
Kemudian Langkah selanjutnya saya akan menghapus salah satu data pelanggan. **DELETE FROM Pelanggan** berguna untuk menghapus record yang ada didalam table **Pelanggan**. **WHERE id_pelanggan = '02'**; adalah lokasi record yang ingin dihapus , disini saya menggunakan primary key **id_pelanggan = '02'** jadi kolom id_pelanggan yang mempunyai record '02' akan dihapus dari table **Pelanggan**.

OUTPUT

- Membuat database baru 'juan_mart'



- Membuat table Pelanggan



- Menghapus kolom kota menggunakan fungsi ALTER

juan_mart/postgres@PostgreSQL 15

Query Query History

```

1 ALTER TABLE Pelanggan DROP COLUMN kota;
2
3 SELECT * FROM Pelanggan;

```

Data Output Messages Notifications

id_pelanggan	nama_pelanggan	umur	alamat
(PK) character varying (20)	character varying (30)	integer	character varying (50)

- Menambahkan 4 buah data record kedalam table Pelanggan

juan_mart/postgres@PostgreSQL 15

Query Query History

```

1 INSERT INTO Pelanggan (id_pelanggan, nama_pelanggan, umur, alamat)
2 VALUES
3 ('01', 'Juan', 20, 'Perumahan Ceunah'),
4 ('02', 'Wirnu', 20, 'Perumahan NutriSari'),
5 ('03', 'Marcelino', 20, 'Perumahan Fortinaiti'),
6 ('04', 'Hasyimadminshope', 21, 'Perumahan Oren');
7
8 SELECT * FROM Pelanggan;

```

Data Output Messages Notifications

	id_pelanggan	nama_pelanggan	umur	alamat
	(PK) character varying (20)	character varying (30)	integer	character varying (50)
1	01	Juan	20	Perumahan Ceunah
2	02	Wirnu	20	Perumahan NutriSari
3	03	Marcelino	20	Perumahan Fortinaiti
4	04	Hasyimadminshope	21	Perumahan Oren

- Mengupdate salah satu data pelanggan

juan_mart/postgres@PostgreSQL 15

Query Query History

```

1 UPDATE Pelanggan SET nama_pelanggan = 'Hasyim'
2 WHERE id_pelanggan = '04';
3
4 SELECT * FROM Pelanggan;

```

Data Output Messages Notifications

	id_pelanggan [PK] character varying (20)	nama_pelanggan character varying (30)	umur integer	alamat character varying (50)
1	01	Juan	20	Perumahan Ceunah
2	02	Wirnu	20	Perumahan NutriSari
3	03	Marcelino	20	Perumahan Fortinaiti
4	04	Hasyim	21	Perumahan Oren

- Menghapus salah satu data pelanggan

juan_mart/postgres@PostgreSQL 15

Query Query History

```

1 DELETE FROM Pelanggan WHERE id_pelanggan = '02';
2
3 SELECT * FROM Pelanggan;

```

Data Output Messages Notifications

	id_pelanggan [PK] character varying (20)	nama_pelanggan character varying (30)	umur integer	alamat character varying (50)
1	01	Juan	20	Perumahan Ceunah
2	03	Marcelino	20	Perumahan Fortinaiti
3	04	Hasyim	21	Perumahan Oren