

BAB II

PEMBUATAN DAN MANAJEMEN TABEL

2.1 Bahasan dan Sasaran

2.1.1 Bahasan

- Pembuatan tabel
- Pengelolaan tabel

2.1.1 Sasaran

1. Mahasiswa memahami cara pembuatan maupun penghapusan tabel.
2. Mahasiswa memahami cara mengedit struktur dari tabel.
3. Mahasiswa mampu memanipulasi tabel sesuai dengan kebutuhan.

2.2 Materi

Sebuah tabel terdiri dari baris (*row*) dan kolom (*column*), dimana jumlah dan nama kolom harus didefinisikan terlebih dahulu di awal. Sedangkan baris merupakan sebuah variabel yang dapat dihapus dan diisi kapanpun, sehingga jumlahnya selalu berubah sesuai dengan jumlah data didalamnya. Setiap kolom dalam tabel mempunyai tipe data, tipe data digunakan untuk membatasi jenis data yang bisa dimasukkan, sehingga akan mempermudah dalam menggunakannya dan melakukan pengelolaan selanjutnya. Sangat disarankan pada saat akan membuat tabel, sebaiknya membuat sebuah aturan khusus dalam penentuan tabel-tabel yang akan dibuat dan memberikan nama kolom maupun tipe datanya.

2.2.1 Membuat Tabel

Catatan :

- Tabel harus memiliki primary key, artinya dalam pembuatan tabel haruslah terdapat sekelompok kolom (*field*) yang menyebabkan setiap baris (*record*) dalam tabel tersebut tidak sama dan juga digunakan untuk mempercepat dalam pencarian.
- Deklarasi primary key tidak boleh kosong (*null*), jadi kita harus mendeklarasikan sebagai *not null*. Namun secara default PostgreSQL menganggapnya sebagai *nullable* (boleh kosong), jika waktu pengisian tidak menyebutkan *null* atau *not null*.

Struktur query yang digunakan membuat tabel dengan dua kolom:

```
create table namatabel (namakolom tipedata keterangan, namakolom tipedata  
keterangan);
```

*Ctn : Untuk melihat struktur tabel yang telah dibuat ketikkan perintah \d dan melihat semua tabel menggunakan \z.

Contoh : Membuat tabel baru dengan nama **infoprib** sebagai berikut :

```
bab2=> create table profil (nip_prof varchar (5) primary key, nama_prof varchar(  
25),kota_prof varchar(15),ttl_prof date, gender_prof char(1));
```

```
CREATE TABLE
```

```
bab2=> \z
```

Schema	Name	Type	Access privileges	Column access privileges
public	profil	table		

<1 row>

```
bab2=> \d profil
```

Column	Table "public.profil" Type	Modifiers
nip_prof	character varying(5)	not null
nama_prof	character varying(25)	
kota_prof	character varying(15)	
ttl_prof	date	
gender_prof	character(1)	

```
Indexes:
```

```
"profil_pkey" PRIMARY KEY, btree (nip_prof)
```

Struktur query yang digunakan menghapus tabel :

```
drop table namatabel;
```

2.2.2 Manipulasi Tabel

Dalam pembuatan tabel sebaiknya direncanakan *field* dan tipe datanya untuk meminimalisir kesalahan saat tabel telah berisi data. Namun meskipun begitu bukan berarti tabel tidak bisa diubah dan dihapus. PostgreSQL telah menyediakan utilitas untuk manipulasi ini dan perlu diingat hal ini bukan manipulasi data melainkan manipulasi struktur tabelnya.

Struktur query yang digunakan untuk menghapus kolom :

```
alter table namatabel drop column "namakolom";
```

Struktur query yang digunakan untuk menambah kolom:

```
alter table namatabel add column "namakolom" tipe data;
```

Struktur query yang digunakan untuk mengubah nama tabel :

```
alter table namatabelasal rename to namatabelbaru;
```

Struktur query yang digunakan untuk mengubah nama kolom :

alter table namatable rename column “namakolom asal” to namakolombaru;

Struktur query yang digunakan untuk mengubah type dari kolom tertentu :

alter table namatable alter column namakolom type namatypepengganti;

Struktur query yang digunakan untuk menambahkan *primary key* pada tabel :

alter table namatable add primary key (namakolom);

2.2.3 Temporary Tabel

Temporary tabel sifatnya hanya sementara artinya akan aktif hanya ketika kita sedang berada atau *login* ke database namun ketika kita *logout* dari *psql* database maka secara otomatis temporary tabel akan terhapus.

Struktur query yang digunakan membuat tabel dengan dua kolom:

**create temporary table namatable (namakolom tipedata keterangan ,
namakolom tipedata keterangan);**

2.2.4 GRANT dan REVOKE

Pada saat membuat tabel pada user tertentu maka hanya user tersebut dan user postgres yang dapat mengakses tabel itu. Namun jika kita ingin agar tabel yang telah dibuat pada user kita dapat diakses oleh user tertentu atau semua user yang berada pada PostgreSQL, maka semua itu dapat dilakukan dengan perintah GRANT. Jika ingin mencabut hak akses kita bisa menggunakan perintah REVOKE.

Berikut struktur penggunaan GRANT untuk semua hak akses :

grant all on table namatable to namauser;

Berikut struktur penggunaan REVOKE untuk semua hak akses :

revoke all on table namatable from namauser;

Struktur query untuk menghapus salah satu hak akses, misalnya UPDATE :

revoke update on tabel namatable from namauser;

Struktur query untuk memberi salah satu hak akses, misalnya DELETE :

grant delete on tabel namatable to namauser;

2.2.5 Turunan (*Inheritance*)

INHERITANCE dipergunakan jika ingin membuat sebuah tabel baru yang berhubungan dengan tabel yang ada, dengan kata lain turunan tabel pertama.

Struktur penggunaan query INHERITANCE :

**create temporary table namatabel (namakolom tipedata keterangan ,
namakolom tipedata keterangan) inherits (namatabelinduk);**

Contoh :

Terdapat tabel kabupaten, kemudian dibuatlah turunan dari tabel kabupaten untuk membuat tabel wisata querynya sebagai berikut :

Create table pariwisata (wisata character varying(20)) inherits (kabupaten) ;

Id_kab	Nama_kab	Desa_kab	
1.	Madiun	Kebonsari	

Id_kab	Nama_kab	Desa_kab	Wisata

Tugas Praktikum

1. Buatlah sebuah tabel dengan nama identitasNIM dan dengan nama fieldnya

Nama	Tipe data	Keterangan
Id_idn	Integer	Not Null
namaDepan_idn	Varchar(10)	
namaBelakang_idn	Varchar(10)	
tgl_lahir_idn	Date	
status_idn	Boolean	
Alamat_idn	Text	

2. Buatlah sebuah tabel lagi dengan nama pegawai dan nama field sama dengan tabel identitas.
3. Hapus kolom “alamat_idn” dan tampilkan struktur dari tabel pegawai tadi.
4. Tambahkan kolom “pekerjaan_idn” dengan *type data varchar* dan tampilkan struktur tabel pegawai.
5. Ubah nama tabel pegawai menjadi pekerjaNIM kemudian ubah nama kolom “pekerjaan_idn” menjadi “alamat_idn” dan tampilkan.
6. Berikan Primary key pada tabel pekerjaNIM dan ubah type data pada kolom “alamat_idn” menjadi *text*. Kemudian tampilkan struktur tabel.
7. Hapus tabel pekerjaNIM.

8. Buat temporary tabel dengan nama **mahasiswa_sementara** dengan 3 kolom yaitu id, nama, tanggal lahir dan tampilkan tabel atau strukturnya.
9. Hapus semua hak akses tabel tertentu kepada salah satu user yang telah dibuat dan lakukan pengecekan. Kemudian berikan semua hak akses kembali dan lakukan pengecekan kembali. Hapus hak akses SELECT tabel tertentu kepada salah satu user dan lakukan pengecekan.
10. Buat tabel baru dengan nama Gaji, dengan catatan pembuatan tabel berasal dari turunan tabel number 1 dan tambahkan kolom gaji kotor, pajak, dan gaji bersih. Kemudian tampilkan strukturnya.

Tugas Rumah

1. Buatlah blog untuk laporan praktikum dan jawaban dari tugas praktikum menggunakan DBMS mysql.
2. Buat database dan tabel sistem pada pertemuan 1 menggunakan query dan dilengkapi dengan tampilan strukturnya.