

Modul Praktikum SQL 3

Basis Data I



**STT TERPADU
NURUL FIKRI**

DML “Data Manipulation Language”

STT Terpadu Nurul Fikri
2018

1. Tampilkan tabel yang sudah dibuat pada database Perusahaan Analog Sentosa

```
mysql> desc pegawai;
```

Field	Type	Null	Key	Default	Extra
ID	int(11)	NO	PRI	NULL	
Nama	varchar(50)	NO		NULL	
Tempat	varchar(30)	NO		NULL	
Tanggal_lahir	date	NO		NULL	
gaji	varchar(35)	NO		NULL	
JK	char(1)	YES		NULL	
Jabatan	varchar(25)	NO		NULL	
Alamat	varchar(40)	NO		NULL	
Nomor_HP	varchar(13)	NO	UNI	NULL	

9 rows in set (0,00 sec)

```
mysql> desc Investor;
```

Field	Type	Null	Key	Default	Extra
ID	int(11)	NO	PRI	NULL	
Nama	varchar(50)	NO		NULL	
jumlah	varchar(35)	NO		NULL	
No_Telp	varchar(15)	NO	UNI	NULL	

4 rows in set (0,08 sec)

```
mysql> desc Proyek;
```

Field	Type	Null	Key	Default	Extra
ID	int(11)	NO	PRI	NULL	
Nama	varchar(45)	NO		NULL	
Tanggal_awal	datetime	NO		NULL	
Tanggal_akhir	datetime	NO		NULL	

4 rows in set (0,08 sec)

Pokok Bahasan

- Membuat database Perusahaan Analog Sentosa
- Pengisian data terhadap database Perusahaan Analog Sentosa
- Pemakaian Data Manipulation Language

Tujuan Praktikum

- Setelah melakukan praktikum mahasiswa diharapkan:

Mengetahui dan mampu membuat dan menggunakan Data Manipulation Language.

Tugas Pendahuluan

- Pastikan environment MySQL anda telah berjalan dengan semestinya (Praktikum 2 SQL)

Percobaan 1 :

1. Amatilah dan lihat perubahan kolom pada tabel pegawai :

Tabel pegawai;

Field	Type	Null	Key	Default	Extra
ID	int(11)	NO	PRI	NULL	
Nama	varchar(50)	NO		NULL	
Tempat	varchar(30)	NO		NULL	
Tanggal_lahir	date	NO		NULL	
gaji	varchar(35)	NO		NULL	
JK	char(1)	YES		NULL	
Jabatan	varchar(25)	NO		NULL	
Alamat	varchar(40)	NO		NULL	
Nomor_HP	varchar(13)	NO	UNI	NULL	
Departemen_id	int(11)	YES		NULL	

10 rows in set (0,00 sec)

- Jika sudah memahaminya, cobalah untuk memasukkan query sesuai dengan analisis yang Anda lakukan.

- Jika sudah ditambahkan, jadikan kolom tersebut sebagai foreign key pada tabel pegawai, jalan kan perintah penambahan foreign key!

- Tampilkan perubahan pada tabel pegawai

2. Amatilah dan lihat perubahan kolom pada tabel Investor :

Tabel Investor

Field	Type	Null	Key	Default	Extra
ID	int(11)	NO	PRI	NULL	
Nama	varchar(50)	NO		NULL	
jumlah	varchar(35)	NO		NULL	
No_Telp	varchar(15)	NO	UNI	NULL	
Proyek_id	int(11)	YES		NULL	
Direktur_id	int(11)	YES		NULL	

6 rows in set (0,17 sec)

- Jika sudah memahaminya, cobalah untuk memasukkan query sesuai dengan analisis yang Anda lakukan.

- Jika sudah ditambahkan, jadikan kolom tersebut sebagai foreign key pada tabel Investor, jalan kan perintah penambahan foreign key!

-Tampilkan Hasil perubahan pada table Investor

Percobaan 2 : Insert data :

1. INSERT DATA Kedalam tabel Pegawai

ID	Nama	Tempat	Tanggal_lahir	gaji	JK	Jabatan	Alamat	Nomor_HP	Departemen_id
1	Nabilah Fajar Utami	Depok	1998-11-24	Rp. 5.300.000	P	Pegawai Keuangan	Cibinong	089662603830	1
2	Raihan Sabiq R	Depok	1998-10-10	Rp. 7.000.000	L	Pegawai Teknologi	Citayam	085779273169	3
3	Muhammad Rizki Herfian	Depok	1998-06-10	Rp. 5.000.000	L	Pegawai Kewirausahaan	Bogor	0895331637524	2
4	Muhammad Syahputra	Bojonggede	1996-10-20	Rp. 3.000.000	L	Pegawai Teknologi	Bojong	083129542930	3
5	Muhammad Naufal Arif W	Depok	1998-05-20	Rp. 8.920.000	L	Pegawai Keuangan	Depok	085888782939	1

5 rows in set (0,00 sec)

- Masukkan data sesuai dengan data yang ada pada tabel Pegawai
- Untuk Kolom tanggal lahir masukkan dengan rules YY/MM/DD

2. INSERT DATA Kedalam tabel Proyek

Tabel Proyek;

ID	Nama	Tanggal_awal	Tanggal_akhir
1	Proyek Database	2018-04-27 00:00:00	2018-04-28 00:00:00
2	Proyek Website	2018-04-28 00:00:00	2018-04-29 00:00:00
3	Proyek Infrastruktur	2018-05-01 00:00:00	2018-06-30 00:00:00
4	Proyek Produk Penambangan	2018-07-01 00:00:00	2018-10-31 00:00:00
5	Proyek Aplikasi Keuangan	2018-06-20 00:00:00	2018-07-10 00:00:00

5 rows in set (0,02 sec)

- Masukkan data sesuai dengan data yang ada pada tabel Proyek
- Untuk Kolom Tanggal awal dan akhir masukkan dengan YY/MM/DD

3. INSERT DATA Kedalam tabel Investor

Tabel Investor;

ID	Nama	jumlah	No_Telp	Proyek_id	Direktur_id
1	Muhammad Fadhil Hilmi	Rp. 500.000.000	089691565266	1	2
2	Muhammad Sholihin	Rp. 100.000.000	085821248290	3	4
3	Ahmad Fathan S	Rp. 250.000.000	082156720429	3	1
4	Ardith Luthfiawan	Rp. 450.000.000	082153912495	5	1
5	Enrico Alkalas	Rp. 700.000.000	083823010249	4	1

5 rows in set (0,00 sec)

- Masukkan data sesuai dengan data yang ada pada tabel Proyek

Percobaan 3 :

1. Ubah Nama Kolom **No_Telp** pada tabel Investor menjadi **Nomer_Telfon** dengan tipe data **varchar** panjang data **13**.
 - Tampilkan perubahan datanya dan jelaskan
2. Ubah Panjang data Kolom gaji pada **tabel Pegawai** menjadi **gaji varchar(30)**
 - Tampilkan perubahan datanya dan jelaskan
3. Hapus Kolom Alamat pada **tabel pegawai**
 - Tampilkan perubahan datanya dan jelaskan
4. Ubah Isi data **Rp. 700.000.000** pada Kolom Jumlah yang ada pada tabel investor menjadi **Rp. 680.000.000**
 - Tampilkan perubahan datanya dan jelaskan
5. Ubah Isi data **2018-07-01** pada Kolom Tanggal Awal yang ada pada tabel Proyek menjadi **2018-05-01**
 - Tampilkan perubahan datanya dan jelaskan
6. Tambah 3 Data pada **Tabel Pegawai**.
 - Tampilkan perubahan datanya dan jelaskan
7. Delete data pada tabel pegawai dengan **kondisi Departemen_id = 3**
 - Tampilkan perubahan datanya dan jelaskan
8. Delete data pada tabel proyek dengan **kondisi id = 3**
 - Tampilkan perubahan datanya dan jelaskan

KAMUS SQL DML

1. **INSERT** → Digunakan untuk menginput data kedalam sebuah tabel.

Penggunaan :

```
INSERT INTO (Nama Tabel) VALUES ("isi kolom1","isi kolom 2",  
"isi kolom 3", . . . , "isi kolom ke-n");
```

2. **SELECT** → digunakan untuk menampilkan sesuatu. Sesuatu di sini bisa berupa sejumlah data dari tabel dan bisa juga berupa suatu ekspresi.

Penggunaan :

```
SELECT [field | *] FROM nama_tabel [WHERE kondisi];
```

3. **ALTER** → Untuk mengubah struktur suatu tabel.

Jenis ALTER dan Penggunaan :

- **ADD** definisi_field_baru Option ini digunakan untuk menambahkan field baru dengan "definisi_field_baru"

```
Alter table (nama table) ADD COLUMN (nama field/kolom,  
tipe dan option lain);
```

```
Alter table (nama table) ADD PRIMARY KEY (nama  
field/kolom, tipe dan option lain);
```

```
Alter table (nama table) ADD FOREIGN KEY (nama  
field/kolom) references nama table induk(primary key  
tabel induk);
```

- **CHANGE** field_yang_diubah definisi_field_baru Option untuk mengubah field_yang_diubah menjadi definisi_field_baru "

```
Alter tabel (nama table) CHANGE (kolom old) (kolom  
new) (kondisi);
```

- **MODIFY** definisi_field Option untuk mengubah suatu field menjadi definisi_field

```
Alter tabel (nama table) MODIFY (Nama Kolom) (NEW Kondisi);
```

- **DROP** nama_field Option untuk menghapus field nama_field
- **RENAME** TO nama_tabel_baru Option untuk mengganti nama tabel

4. **DELETE** → Proses delete dilakukan jika terdapat data atau record dalam suatu tabel yang perlu dihapus atau dihilangkan. Perubahan yang terjadi dalam proses delete bersifat permanen, artinya setelah perintah dijalankan tidak dapat di-cancel (undo). Jadi berhati-hatilah dengan perintah delete !

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

5. **UPDATE** → Proses update bisa sewaktu-waktu dilakukan jika terdapat data atau record dalam suatu tabel yang perlu diperbaiki. Proses update ini tidak menambahkan data (record) baru, tetapi memperbaiki data yang lama. Perubahan yang terjadi dalam proses update bersifat permanen, artinya setelah perintah dijalankan tidak dapat di-cancel (undo).

```
UPDATE nama_tabel SET field1='nilaibaru' [WHERE kondisi];
```