



Basis Data (DataBase) ⁷

Hanhan Hanafiah Solihin, S.Kom., M.Kom
Universitas Sangga Buana

Mengenal MySQL

- MySQL adalah program database server yang mampu menerima dan mengirimkan datanya dengan sangat cepat, multi user serta menggunakan perintah standar SQL (Structured Query Language)
- MySQL adalah sebuah database yang dapat digunakan baik sebagai client maupun server
- Database MySQL merupakan sebuah perangkat lunak database yang berbentuk database relasional atau dalam bahasa basis data disebut sebagai Relational Database Management System (RDBMS) yang menggunakan suatu bahasa permintaan SQL

Kelebihan MySQL

- MySQL sebagai Database Management System (DBMS)
- MySQL sebagai Relational Database Management System (RDBMS)
- MySQL adalah sebuah software database yang opensource, artinya program ini bersifat free atau bebas digunakan oleh siapa saja tanpa harus membeli dan membayar lisensi kepada pembuatnya
- MySQL merupakan sebuah database server yang melayani permintaan dari client
- MySQL merupakan sebuah database client yang dapat melakukan query yang mengakses database pada server
- MySQL mampu menerima query yang bertumpuk dalam satu permintaan atau disebut Multi-Threading

Kelebihan MySQL

- MySQL merupakan sebuah database yang mampu menyimpan data berkapasitas sangat besar hingga berukuran gigabyte sekalipun
- MySQL didukung oleh driver ODBC, artinya database MySQL dapat diakses menggunakan aplikasi apa saja termasuk berupa visual seperti Delphi maupun Visual Basic, VB •Net, Java
- MySQL adalah database yang menggunakan enkripsi password
- MySQL merupakan server database yang multi user artinya database ini dapat digunakan oleh banyak orang
- MySQL dapat menciptakan lebih dari 16 kunci per tabel dan dalam satu kunci memungkinkan berisi belasan field

Kelebihan MySQL

- MySQL mendukung field yang dijadikan sebagai kunci primer dan kunci unique
- MySQL didukung oleh sebuah komponen C dan perl API, sehingga dapat diakses melalui sebuah program aplikasi yang berada di bawah protokol internet berupa web
- MySQL menggunakan suatu bahasa permintaan standar yang bernama SQL

Perbedaan MySQL dan SQL

- **MySQL** adalah program database server sedangkan SQL adalah bahasa yang digunakan didalamnya
- **SQL** dibedakan menjadi dua bentuk query yaitu :
 - **DDL** (*Data Definition Language*) adalah sebuah metode query SQL yang berguna untuk mendefinisikan data pada sebuah database yaitu :
 - ✓ **CREATE**, digunakan untuk melakukan pembuatan database dan tabel. Contoh : CREATE DATABASE, CREATE TABLE
 - ✓ **DROP**, digunakan untuk melakukan penghapusan tabel maupun database. Contoh : DROP TABLE, DROP DATABASE
 - ✓ **ALTER**, digunakan untuk melakukan pengubahan struktur tabel yang telah dibuat, baik menambah field (add), mengganti nama field (change) ataupun menamakannya kembali (rename) serta menghapus (drop)

Perbedaan MySQL dan SQL

- DML (*Data Manipulation Language*) adalah sebuah metode query SQL yang digunakan apabila DDL telah terjadi, sehingga fungsi dari query ini adalah untuk melakukan pemanipulasian database yang telah ada atau telah dibuat sebelumnya, yaitu :
 - ✓ **INSERT**, digunakan untuk melakukan pemasukan data pada tabel database
 - ✓ **UPDATE**, digunakan untuk melakukan pengubahan terhadap data yang ada pada tabel
 - ✓ **DELETE**, digunakan untuk melakukan penghapusan data pada tabel

Perintah DML

- DML = *Data Manipulation Language*
- Digunakan untuk memanipulasi data
- Contoh:
 - SELECT – mengambil data
 - DELETE – menghapus data
 - INSERT – menyisipkan data
 - UPDATE – mengubah data

Tipe Data MySQL

Data Numerik

Tipe	Byte	Keterangan
TINYINT	1	Menangani data mulai dari angka -128 s.d. 127
SMALLINT	2	Menangani data mulai dari angka -32768 s.d. 32767
MEDIUMINT	3	Menangani data mulai dari angka -8388608 s.d. 8388607
INT	4	Menangani data mulai dari angka -2147483648 s.d. 2147483647
BIGINT	8	Menangani data mulai dari angka -9223372036854775808 s.d. 9223372036854775807

Type Data MySQL

Data Teks

Type	Byte	Keterangan
TINYTEXT	255	Menangani data sampai dengan 2^8-1 data
TINYLOB	255	Menangani data sampai dengan 2^8-1 data
TEXT	65535	Menangani data sampai dengan $2^{16}-1$ data
BLOB	65535	Menangani data sampai dengan $2^{16}-1$ data
MEDIUMTEXT	16777215	Menangani data sampai dengan $2^{24}-1$ data
MEDIUMBLOB	16777215	Menangani data sampai dengan $2^{24}-1$ data
LOBLOB	4294967295	Menangani data sampai dengan $2^{32}-1$ data

Tipe Data MySQL

Data String

Tipe	Keterangan
VARCHAR	Menangani data hingga 225 karakter
CHAR	Menangani data sampai dengan 225 karakter Perbedaan antara varchar dengan char adalah jika Anda membuat kolom dengan tipe varchar(25) maka Anda dapat memasukkan data paling banyak atau kurang dari 25 digit , sedangkan jika Anda membuat kolom dengan tipe char(4) maka Anda harus mengisikan data paling banyak 4 karakter dan paling sedikit 4 karakter saja, sehingga apabila data yang dimasukkan sebanyak 2 karakter maka data tersebut tetap dibaca dalam 4 digit
ENUM	Tipe ini disebut sebagai tipe validasi. Pada tipe ini data yang akan mengisi kolom harus ditentukan terlebih dahulu. Misalnya, pada pembuatan kolom golongan darah yang memiliki anggota A,B,AB dan O, maka bentuk penulisannya adalah ENUM('A','B','AB','O')
SET	Memiliki fungsi yang sama dengan ENUM

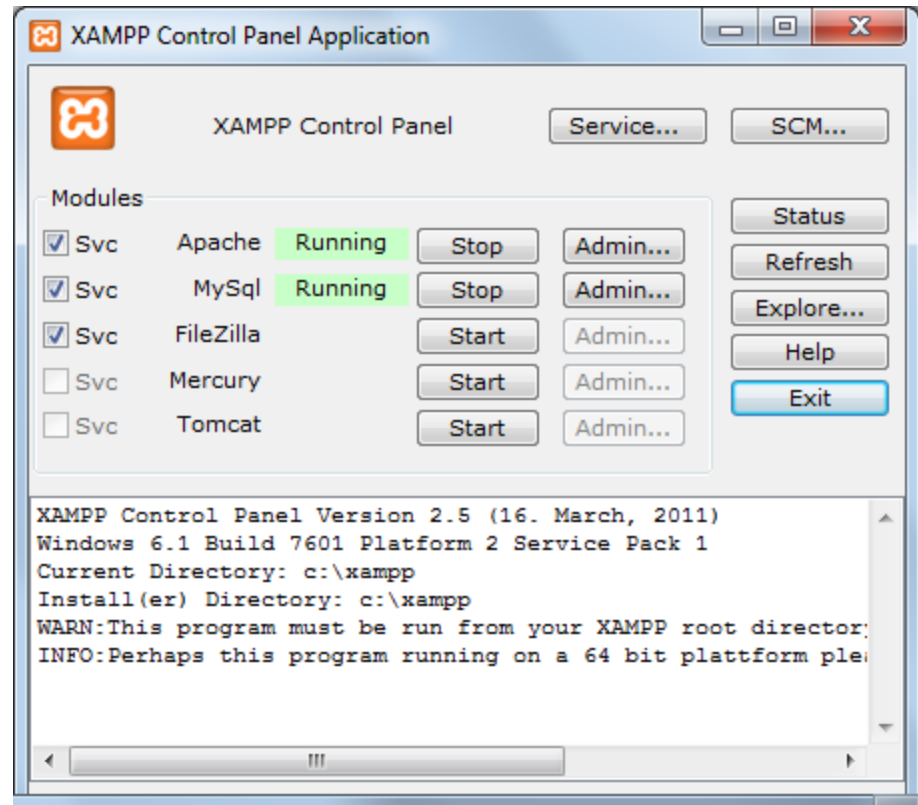
Tipe Data MySQL

Data Penanggalan dan Waktu

Tipe	Keterangan
DATETIME	Menangani data penanggalan dan waktu dengan bentuk '0000-00-00 00:00:00', dimana 0000-00-00 merupakan bentuk penanggalan tahun-bulan-tanggal dan 00:00:00 merupakan bentuk waktu jam:menit:detik
DATE	Menangani data penanggalan dengan bentuk '0000-00-00'
TIMESTAMP	Menangani data penanggalan yang diciptakan berjajar tanpa ada pembatasnya dengan prototipe penulisannya 0000000000000000
TIME	Menangani data waktu dengan bentuk '00:00:00'
YEAR	Menangani data tahun yang ditulis secara lengkap 4 digit

Mengkatifkan MySQL

- Dalam praktikum kali ini tools yang akan digunakan adalah XAMPP
- XAMPP adalah aplikasi yang sudah memiliki fasilitas MySQL didalamnya.
- Syarat pertama untuk menjalankan MySQL di XAMPP adalah memastikan MySQL di Control panel XAMPP telah aktif.



Mengkatifkan MySQL

- Buka Command Prompt Atau CMD pada run.
- Ketikan `cd c:\xampp\mysql\bin` → **ENTER**
- Ketikan `mysql -uroot` → **ENTER**

(Jika memiliki password gunakan juga `-uroot -p` → **ENTER** dan ketikan password)

* **root** adalah pemakai dengan wewenang paling tinggi dan digunakan untuk admin sistem

Mendefinisikan Database

- Menciptakan database baru :

CREATE DATABASE nama_database;

- Menampilkan seluruh database yang ada di direktori Data :

SHOW DATABASES;

- Mengaktifkan database :

USE nama_database;

- Menghapus database yang ada :

DROP DATABASE nama_database;

- **NOT NULL** : Tidak boleh kosong
- **PRIMARY KEY** : Kunci primer

Mendefinisikan Tabel

- Mendefinisikan Tabel

```
CREATE TABLE nama_tabel  
(field-1 type(length), field-2 type(length),  
...., field-n type(length));
```

- Menciptakan tabel baru dengan kolom tidak boleh kosong :

```
CREATE TABLE nama_tabel  
(field-1 type(length) NOT NULL, ....,  
field-n type(length));
```


Mendefinisikan Tabel

- Menciptakan tabel baru dengan primary key :

```
CREATE TABLE nama_tabel  
(kolom_primer type(length) PRIMARY KEY, .....,  
field-n type(length));
```

- Atau

```
CREATE TABLE nama_tabel  
(kolom_primer type(length), ....., field-n  
type(length), PRIMARY KEY(kolom_primer));
```

Contoh :

```
create database AKADEMIK;
```

```
create TABLE mahasiswa(  
  nomor int(3) not null,  
  npm int(9) not null primary key,  
  nama varchar(25) not null,  
  kelas char(8));
```

Mendefinisikan Tabel

- Menampilkan tabel-tabel yang ada dalam database :

SHOW TABLES;

- Melihat struktur tabel :

DESC nama_tabel;

- Menghapus tabel :

DROP TABLE nama_tabel;

Memanipulasi Tabel

- Mengganti nama tabel :

RENAME TABLE tabel_lama TO tabel_baru;

- Menambah kolom ke dalam tabel :

ALTER TABLE nama_tabel ADD nama_field type(length);

- Mengubah kolom menjadi kunci primer :

ALTER TABLE nama_tabel ADD PRIMARY KEY (nama_kolom);

- Menghapus kunci primer :

ALTER TABLE nama_tabel DROP PRIMARY KEY;

Memanipulasi Tabel

- Mengubah nama dan tipe kolom :

ALTER TABLE nama_tabel CHANGE kolom_lama kolom_baru type(length);

- Menghapus kolom :

ALTER TABLE nama_tabel DROP nama_field;

Memasukkan Data ke Tabel

- Memasukkan data ke semua field :

```
INSERT INTO nama_tabel  
VALUES ('isi field1','isi field2','...','isi fieldn');
```

- Memasukkan data ke field-field tertentu :

```
INSERT INTO nama_tabel (field1,field2,...,fieldn)  
VALUES ('isi field1','isi field2','...','isi fieldn');
```

Memasukkan Data ke Tabel

- Memasukkan data ganda ke semua field :
INSERT INTO nama_tabel VALUES
(‘isi field1’, ‘isi field2’, ‘...’, ‘isi fieldn’),
(‘.....’, ‘.....’, ‘...’, ‘.....’),
(‘isi field1’, ‘isi field2’, ‘...’, ‘isi fieldn’);
- Memasukkan data ganda ke field-field tertentu :
INSERT INTO nama_tabel
(field1, field2, ..., field n) VALUES
(‘isi field1’, ‘isi field2’, ‘...’, ‘isi fieldn’),
(‘.....’, ‘.....’, ‘...’, ‘.....’),
(‘isi field1’, ‘isi field2’, ‘...’, ‘isi fieldn’);

Contoh :

```
INSERT into mahasiswa (nomor,npm,nama,kelas)
values
('01', '10507498', 'Angga', 'USB-1'),
('02', '10507482', 'Ginanmar', 'USB-1'),
('03', '10507503', 'Hari', 'USB-1'),
('04', '10507496', 'Yayan', 'USB-1'),
('05', '10507516', 'Arifki', 'USB-1');
```


Menampilkan Data dari Tabel

- Menampilkan isi seluruh tabel :
SELECT * FROM nama_tabel;
- Menampilkan isi seluruh tabel secara berurutan menaik sesuai kolom tertentu
**SELECT * from nama_tabel ORDER BY
nama_kolom asc;**
- Menampilkan isi seluruh tabel secara berurutan menurun sesuai kolom tertentu
**SELECT * from nama_tabel ORDER BY
nama_kolom desc;**
- Menampilkan isi field-field tertentu dari tabel :
SELECT field1, field2, ..., fieldn FROM nama_tabel;
- Menampilkan isi field-field tertentu dari tabel dengan memberikan judul untuk kolom dengan spasi :
SELECT nama_kolom AS 'judul' FROM nama_tabel;

Memperbaharui Isi Data

- Memperbaharui isi tabel tanpa kondisi :

UPDATE nama_tabel SET

field1='data_baru',field2='data_baru',...,fieldn='data_baru';

- Memperbaharui isi tabel dengan kondisi :

UPDATE nama_tabel SET

field1='data_baru',field2='data_baru',...,fieldn='data_baru' WHERE [kondisi];

Menghapus Isi Data

- Menghapus isi tabel tanpa kondisi :

DELETE FROM nama_tabel;

Menghapus isi tabel dengan kondisi :

DELETE FROM nama_tabel WHERE [kondisi];

Terimakasih