



PENGANTAR MYSQL

- Pendahuluan
- Keunggulan MySQL
- Sistem Server Database MySQL
- Referensi

Pendahuluan

SQL (Structured Query Language) adalah bahasa stndart yang digunakan untuk mengakses server database . Semenjak tahun 70-an bahasa ini telah dikembangkan oleh IBM, yang kemudian diikuti dengan adanya Oracle, Informix dan Sybase. Dengan menggunakan SQL, proses akses database menjadi lebih user-friendly dibandingkan dengan misalnya dBase ataupun Clipper yang masih menggunakan perintah – perintah pemrograman murni.

Selain MySQL, ada beberapa jenis pemograman yang berorientasi database yang dapat digunakan untuk aplikasi di web seperti ORACLE. Oracle merupakan sebuah perusahaan besar di dunia yang cakupan bisnis salah satunya adalah penjualan software dan pembuatan software database yang diperuntukkan bagi perusahaan-perusahaan besar di dunia. Karena softwarena tidak bebas di “pasaran” atau tidak free software maka sebagian besar perusahaan kecil atau menengah atau programmer web masih menggunakan database MySQL sebagai software database perusahaan atau webnya.

Berita terayar mengabarkan bahwa ORACLE merupakan perusahaan yang mendapatkan keuntungan terbesar dalam bisnis software database.

MySQL adalah sebuah server database SQL multiuser dan multi-threaded. SQL sendiri adalah salah satu bahasa database yang paling populer di dunia. Implementasi program server database ini adalah program daemon 'mysqld' dan beberapa program lain serta beberapa pustaka.

MySQL dibuat oleh TcX dan telah dipercaya mengelola sistem dengan 40 buah database berisi 10,000 tabel dan 500 di antaranya memiliki 7 juta baris (kira-kira 100 gigabyte data). Database ini dibuat untuk keperluan sistem database yang cepat, handal dan mudah digunakan. Walaupun memiliki kemampuan yang cukup baik, MySQL untuk sistem operasi Unix bersifat freeware, dan terdapat versi shareware untuk sistem operasi windows. Menurut pembuatnya, MySQL disebut seperti "my-ess-que-ell" dan bukan my-sequel !

Sebagaimana database sistem yang lain, dalam SQL juga dikenal hierarki server dengan database-database. Tiap-tiap database memiliki tabel-tabel. Tiap-tiap tabel memiliki field-field.

Umumnya informasi tersimpan dalam tabel – tabel yang secara logik merupakan struktur 2 dimensi terdiri atas baris dan kolom. Field-field tersebut dapat berupa data seperti int , realm char, date, time dan lainnya.

SQL tidak memiliki fasilitas pemrograman yang lengkap, tidak ada looping ataupun percabangan ,misalnya. Sehingga untuk menutupi kelemahan ini perlu digabung dengan bahasa pemrograman semisal C.

Dalam training ini kita menggunakan MySQL sebagai SQL server karena berbagai kelebihannya. Antara lain;

1. Source MySQL dapat diperoleh dengan mudah dan gratis
2. Sintaksnya lebih mudah dipahami dan tidak rumit
3. Pengaksesan database dapat dilakukan dengan mudah

Keunggulan MySQL

- MySQL merupakan program yang multi-threaded, sehingga dapat dipasang pada server yang memiliki multi-CPU.
- Didukung program-program umum seperti C, C++, Java, Perl, PHP, Python, TCL APIs dls.

- Bekerja pada berbagai platform. (tersedia berbagai versi untuk berbagai sistem operasi).
- Memiliki jenis kolom yang cukup banyak sehingga memudahkan konfigurasi sistem database.
- Memiliki sistem sekuriti yang cukup baik dengan verifikasi host.
- Mendukung ODBC untuk sistem operasi Microsoft Windows.
- Mendukung record yang memiliki kolom dengan panjang tetap atau panjang bervariasi. dan masih banyak keunggulan lainnya (lihat manual mysql_3.22)
- MySQL merupakan software yang free, dan bisa di download di www.mysql.com. Sedangkan software database lainnya seperti ORACLE merupakan software yang harus di beli.
- MySQL dan PHP saling terintegrasi. Maksudnya adalah pembuatan database dengan menggunakan sintak PHP dapat di buat. Sedangkan input yang di masukkan melalui aplikasi web yang menggunakan script server-side seperti PHP dapat langsung dimasukkan ke database MySQL yang ada di server dan tentunya web tersebut berada di sebuah web server.

Sistem Server Database MySQL

Sistem database MySQL memiliki sistem sekuritas dengan tiga verifikasi yaitu user(name), password dan host. Verifikasi host memungkinkan untuk membuka sekuriti di 'localhost', tetapi tertutup bagi host lain (bekerja di lokal komputer). Sistem sekuriti ini ada di dalam database mysql dan pada tabel

user. Proteksi juga dapat dilakukan terhadap database, tabel, hingga kolom secara terpisah.

Referensi

Referensi pengetahuan tentang MySQL secara umum telah dipandang cukup dari manual yang telah diberikan oleh TcX (mysql_3.22). Beberapa buku lain tentang database SQL banyak dijumpai dan dapat memberikan pengetahuan tambahan tentang pemrograman SQL.

Akses Databases

Pada umumnya akses ke database melalui tiga tahapan, yaitu :

1. Koneksi ke database (persiapan)
- 2 .Query/permintaan data (operasi)
3. Pemutusan koneksi

Koneksi ke database dilakukan menggunakan fungsi-fungsi *mysql_connect()*, *mysql_pconnect()*, *mysqlselect_db()*.

Untuk lebih lanjutnya, kita akan membahasnya pada bagian modul berikutnya.

INSTALASI DAN TIPE DATA

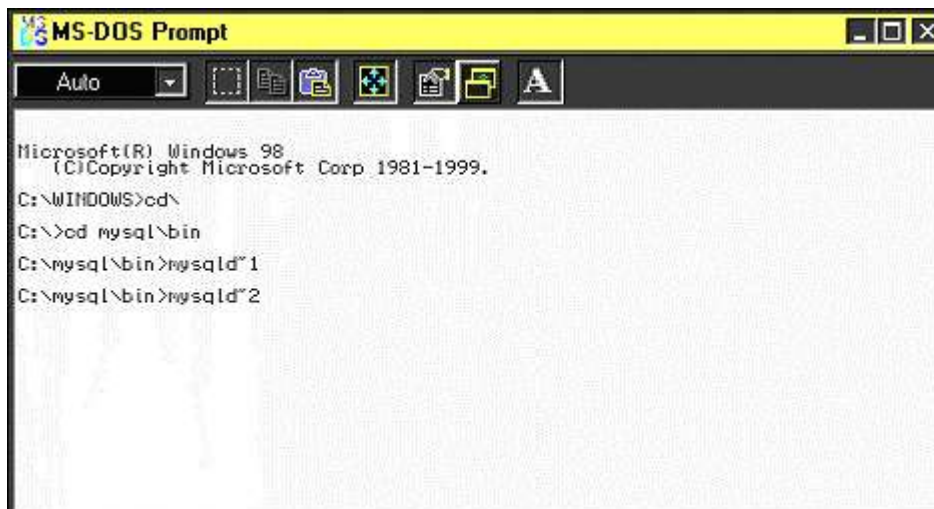
- Instalasi MySQL
- Tipe Data

Instalasi MySQL

Untuk bisa bekerja dengan software MySQL, tentunya software tersebut harus terinstal terlebih dahulu di komputer kita. Untuk mendapatkan software tersebut kita dapat download dari website resmi MySQL yaitu <http://www.mysql.com>. Di website tersebut ada beberapa software MySQL dari yang terlama sampai yang terbaru.

Dalam modul ini kita akan menggunakan MySQL versi mysql-3.23.32-win. Setelah anda download, file tersebut harus di ekstrak supaya kita dapat menginstalnya. Setelah di ekstrak maka kita install dengan menekan file *setup* dua kali. Selanjutnya ikuti instruksi yang ada ketika instalasi.

Setelah kita berhasil menginstal software tersebut, maka untuk menjalankannya kita harus memulainya dari Dos-Prompt. Buka tampilan Dos-Prompt dari menu program pada tombol start. Jika kita meletakkan file instalasi mysql di direktori “c:\mysql” maka pada tampilan Dos-Prompt kita masuk ke direktori tersebut. Perhatikan gambar di bawah ini :



```
MS-DOS Prompt
Auto
Microsoft(R) Windows 98
(C)Copyright Microsoft Corp 1981-1999.
C:\WINDOWS>cd\
C:\>cd mysql\bin
C:\mysql\bin>mysqld"1
C:\mysql\bin>mysqld"2
```

Dari gambar di atas, maka dapat dijelaskan bahwa :

Pada baris pertama yaitu *c:\windows\cd*

Dengan di tulisnya *cd* berarti kita pindah ke directori asal yaitu “c:\”. Ada juga sintak “*cd..*” yang berarti bahwa kita naik ke folder yang di atas folder sebelumnya.

Karena ketika kita menginstalnya di folder *c:\mysql* maka kita harus masuk ke *mysql*. Untuk mengkonfigurasinya maka kita harus masuk ke folder “*bin*” dengan menuliskan pada tampilan Dos-Prompt “*c:\mysql\bin*”

Agar modul *mysql* berjalan dengan baik dan dapat terakses dengan benar maka kita harus menjalankan *mysqldump* yang ada di folder *bin* dengan sintak *mysqld~2*, juga kita harus menjalankan *mysql-opt* dengan sintak *mysqld~1*.

Agar bisa MySQL dapat berjalan dengan baik di mana software tersebut terjaga dengan diberi password dan password tersebut terestriksi dengan baik maka gunakan sintak :

```
MS-DOS Prompt
Auto
--fields-optionally-enclosed-by=...
    Fields in the i.file are opt. enclosed by ...
--fields-escaped-by=...
    Fields in the i.file are escaped by ...
--lines-terminated-by=...
    Lines in the i.file are terminated by ...

Default options are read from the following files in the given order:
C:\WINDOWS\my.ini c:\my.cnf
The following groups are read: mysqldump client
The following options may be given as the first argument:
--print-defaults      Print the program argument list and exit
--no-defaults        Don't read default options from any options file
--defaults-file=#     Only read default options from the given file #
--defaults-extra-file=# Read this file after the global files are read

Possible variables for option --set-variable (-O) are:
max_allowed_packet  current value: 25165824
```

Sintak *mysqladmin reload* merupakan perintah untuk mengaktifkan mysql dan agar user dan passwordnya dapat aktif dengan baik.

Sintak *mysqladmin -u root password smkti* berarti nama user adalah root dan password-nya adalah *smkti*.

Hal ini sangat berguna ketika kita ingi menghubungkan sintak PHP dengan MySQL karena untuk menghubungkannya kita membutuhkan nama user yang memiliki database tertentu dengan password yang harus terproteksi supaya user lain tidak dapat melihat dan mengganggu ini database yang telah ada.

Langkah selanjutnya adalah :

```
MYSQL
Auto
C:\WINDOWS\my.ini c:\my.cnf
The following groups are read: mysqldump client
The following options may be given as the first argument:
--print-defaults      Print the program argument list and exit
--no-defaults        Don't read default options from any options file
--defaults-file=#     Only read default options from the given file #
--defaults-extra-file=# Read this file after the global files are read

Possible variables for option --set-variable (-O) are:
max_allowed_packet  current value: 25165824
net_buffer_length   current value: 1047551

C:\mysql\bin>mysqladmin reload

C:\mysql\bin>mysqladmin -u root password smkti

C:\mysql\bin>mysql -u root -p
Enter password: *****
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 3 to server version: 3.23.32

Type 'help;' or '\h' for help. Type '\c' to clear the buffer

mysql>
mysql>
```

Untuk bisa masuk kedalam system MySQL maka kita harus menuliskan sintak :

```
mysql -u root -p
```

Maka akan muncul kalimat *enter password* . Password yang kita masukkan adalah password yang sudah kita nyatakan pada sintak sebelumnya yaitu *smkti*.

Jika telah nampak sintak :

```
mysql>
```

```
mysql>
```

maka kita telah masuk kedalam system database MySQL tersebut.

Tipe Data

Beberapa jenis tipe data dalam MySQL memiliki tipe tersendiri dalam tiap field di table databasenya.

MySQL mengenal beberapa type data field, yaitu :

1. Tipe data numerik

Tipe data numerik dapat dibedakan menjadi dua kelompok, yaitu tipe data integer dan tipe data floating point. Tipe data integer untuk data bilangan bulat sedangkan tipe data floating point digunakan untuk bilangan desimal.

Tipe data numeric dapat kita sajikan dalam table di bawah ini :

Tipe data	Kisaran nilai
Tinyint	(-128) – 127 atau 0-225
Smallint	(-32768) – 32767 atau 0 – 65535
Mediumint	(-8388608)-8388607 atau 0-16777215
Int	(-2147683648)-(-2147683647) atau 0-4294967295
Bigint	(-9223372036854775808)-(-9223372036854775807) atau 0 - 18446744073709551615
Float(x)	(-3.402823466E+38)-(-1.175494351E-38),0,dan 1.175494351E-38 – 3.402823466E + 38
Float	Idem
Double	(-1.7976E+308)-(-2.22E-308),0,dan(2.22E-308)- (1.79E+308)

Besar kebutuhan memori penyimpanan untuk masing-masing tipe data di atas sebagai berikut :

Column type	Storage required
TINYINT	1 byte
SMALLINT	2 bytes
MEDIUMINT	3 bytes
INT	4 bytes
INTEGER	4 bytes
BIGINT	8 bytes
FLOAT(4)	4 bytes
FLOAT(8)	8 bytes
FLOAT	4 bytes
DOUBLE	8 bytes
DOUBLE PRECISION	8 bytes
REAL	8 bytes
DECIMAL(M,D)	M bytes (D+2, if M < D)
NUMERIC(M,D)	M bytes (D+2, if M < D)

2. Tipe data string

Yang termasuk dalam tipe data string adalah tipe-tipe data berikut :

Tipe kolom	Kebutuhan memori penyimpanan
CHAR(M)	M bytes, $1 \leq M \leq 255$
VARCHAR(M)	L+1 bytes, where $L \leq M$ and $1 \leq M \leq 255$
TINYBLOB, TINYTEXT	L+1 bytes, where $L < 2^8$
BLOB, TEXT	L+2 bytes, where $L < 2^{16}$
MEDIUMBLOB, MEDIUMTEXT	L+3 bytes, where $L < 2^{24}$
LOBLOB, LONGTEXT	L+4 bytes, where $L < 2^{32}$
ENUM ('value1','value2',...)	1 or 2 bytes, depending on the number of enumeration values (65535 values maximum)
SET ('value1','value2',...)	1, 2, 3, 4 or 8 bytes, depending on the number of set members (64 members maximum)

3. Tipe data waktu

Yang termasuk dalam tipe data tanggal dan waktu adalah sebagai berikut :

Column type	Storage required
DATETIME	8 bytes
DATE	3 bytes
TIMESTAMP	4 bytes
TIME	3 bytes
YEAR	1 byte

Tipe data `char()` dan `varchar()` pada prinsipnya sama. Yang menjadi perbedaannya adalah pada jumlah memori yang dibutuhkan untuk penyimpanan. Memori penyimpanan yang dibutuhkan tipe data `char()` bersifat statis, besarnya tergantung pada berapa jumlah karakter yang ditetapkan pada saat field tersebut dideklarasikan. Sebaliknya, tipe data `varchar()` besar memori penyimpanan tergantung terhadap berapa karakter yang digunakan ditambah 1 byte yang berisi data jumlah karakter yang digunakan.

SINTAK DASAR MySQL

- Pengantar Sintak Dasar
- Membuat Database Baru

Pengantar Sintak Dasar

MySQL merupakan bahasa pemrograman database di mana penulisan sintaknya tidak serumit bahasa pemrograman lainnya seperti java, C++ dan sebagainya. Satu hal yang perlu diingat bahwa setiap penulisan script MySQL di Dos-Prompt harus selalu diakhiri dengan tanda titik koma (;).

Di dalam source MySQL yang telah terinstal secara default telah terisi sebuah database yang bernama *mysql* dan *tes* . Untuk dapat menampilkan apasaja nama database yang telah ada maupun yang akan kita buat, gunakan sintak :

```
Mysql> show databases;
```

Perhatikan contoh di bawah ini :

```
mysql>
mysql> show databases;
+-----+
| Database |
+-----+
| mysql    |
| test     |
+-----+
2 rows in set (0.11 sec)
mysql> _
```

Contoh di atas menunjukkan bahwa dengan menggunakan sintak *show databases;* berarti kita dapat menampilkan seluruh nama database yang telah ada. Sintak untuk bisa masuk kedalam salah satu system database tersebut adalah :

```
Mysql> use nama_database;
```

Perhatikan gambar di bawah ini :

```
mysql>
mysql> show databases;
+-----+
| Database |
+-----+
| mysql    |
| test     |
+-----+
2 rows in set (0.11 sec)
mysql> use mysql;
Database changed
mysql>
```

Dengan mengetikkan sintak *use mysql;* dan keluar kalimat database change, berarti kita telah masuk ke dalam database mysql yang telah ada.

Di setiap database tentu ada terdapat beberapa table yang menjadi komponen dasar sebuah database. Sintak untuk menampilkan seluruh table yang telah ada di dalam database mysql adalah :

```
mysql>show tables;
```

perhatikan gambar di bawah :

```
mysql>
mysql> use mysql;
Database changed
mysql> show tables;
+-----+
| Tables_in_mysql |
+-----+
| columns_priv    |
| db              |
| host            |
| tables_priv     |
| user            |
+-----+
5 rows in set (0.22 sec)

mysql>
```

Dari gambar di atas, kita dapat melihat bahwa di dalam database mysql sudah ada beberapa table yang telah default dari mysql-nya sendiri. Didalam database tersebut ada table *columns_priv* (*columns_privilege*), *db*, *host*, *tables_privilege*, dan *user* di mana tiap tablenya memiliki fungsi tertentu yang dapat diakses oleh tiap user. Database mysql ini biasanya sering digunakan bagi webmaster sebagai tempat penyimpanan data, karena table-table yang telah ada dan table yang dibutuhkan sebagian besar telah terpenuhi. Fungsi tiap table tersebut akan kita bahas di bawah ini.

Untuk bisa melihat isi seluruhnya dari salah satu table seperti user, maka gunakan sintak sebagai berikut :

```
mysql>select*from user;
```

Maka akan tampil suatu kolom-kolom dan baris-baris yang telah ada seperti gambar di bawah ini:

```

mysql>
+-----+-----+-----+-----+-----+-----+-----+-----+
| Host      | User | Password          | Select_priv | Insert_priv | Update_priv |
| Delete_priv | Create_priv | Drop_priv | Reload_priv | Shutdown_priv | Process_
priv | File_priv | Grant_priv | References_priv | Index_priv | Alter_priv |
+-----+-----+-----+-----+-----+-----+-----+-----+
| localhost | root | 1fb744464c9bba54 | Y           | Y           | Y           |
| Y         | Y     | Y                 | Y           | Y           | Y           |
| %         |      |                   | N           | N           | N           |
| N         | N     | N                 | N           | N           | N           |
| localhost |      |                   | Y           | Y           | Y           |
| Y         | Y     | Y                 | Y           | Y           | Y           |
| %         | root |                   | Y           | Y           | Y           |
| Y         | Y     | Y                 | Y           | Y           | Y           |
+-----+-----+-----+-----+-----+-----+-----+
4 rows in set (2.36 sec)
mysql>

```

Dari gambar di atas akan terlihat kolom host, user dan password yang telah terisi. Isi dalam kolom tersebut seperti :

Host	User	Password
Localhost	Root	1fb744464c9bba54

Localhost merupakan nama komputer kita sendiri apabila komputer tersebut kita jadikan server. Sedangkan root adalah nama user yang berhak penuh untuk mengatur database yang telah ada seperti menambah user, memberi fasilitas kepada user lain agar user lain dapat akses ke database tersebut, bahkan juga root bisa membuat database baru yang dapat digunakan kapanpun. Kolom password yang terisi oleh kombinasi k'angka dan huruf di atas yaitu 1fb744464c9bba54 merupakan hasil password yang telah terestriksi pada sintak terdahulu yaitu dengan menambahkan sintak *password* di depan kata password itu sendiri.

Membuat Database Baru

Membuat database

Untuk membuat sebuah database dengan nama 'formulir' kita tidak perlu harus keluar terlebih dahulu dari salah satu database walau kita sebelumnya telah masuk ke dalamnya, untuk membuatnya gunakan syntax berikut

[mysql>create database formulir;](#)

Perhatikan gambar di bawah ini :

```
mysql> select * from host;
Empty set (0.05 sec)

mysql> create database formulir;
Query OK, 1 row affected (0.88 sec)

mysql>
```

Jika ada kalimat *query ok, 1 row affected (0,88 sec)* berarti kita telah berhasil membuat sebuah database yang bernama *formulir*.

Dan untuk menghapusnya, gunakan sintak sebagai berikut :

[mysql>drop database formulir;](#)

Perhatikan gambar di bawah ini :

```
mysql> drop database formulir
-> ;
Query OK, 0 rows affected (0.72 sec)

mysql>
```

Pada kondisi di atas, apabila kita lupa meletakkan tanda titik koma (;) di akhir kalimat maka pada baris setelahnya akan keluar tanda (->). Di samping tanda tersebut harus kita beri tanda titik koma (;) agar dapat berjalan dengan baik.

Membuat tabel

Setelah kita memasuki sebuah database, kita dapat mulai membuat tabel – tabel sesuai dengan keperluan kita. Dalam database dikenal primary key, yaitu field yang menjadi acuan data terhadap field-field lainnya dan primary key tidak boleh null. Misalkan nama tabelnya adalah 'data', maka sintaknya adalah :

[mysql>create table data \(field1 typefield1 not null , field2 typefield2, primary key \(field1\)\);](#)

contoh :

```
create table data (nama char(15) not null, alamat char(20), primary key(nama));
```

Perhatikan gambar di bawah ini :

```
mysql> create table data (nama char(15) not null, alamat char(20), primary key(nama));
Query OK, 0 rows affected (0.44 sec)
mysql>
```

Berbeda dengan sintak membuat database yang baru (*create database formulir*), membuat table dalam database harus langsung disertai dengan menuliskan nama kolom dan tipe datanya. Tanpa membuat hal tersebut maka kita belum bis membuat suatu table karena kolomnya belum ada. Dan juga untuk membuat suatu table yang baik maka harus disertai dengan *primary key* sebagai kolom pembeda dari kolom yang lainnya agar user dalam mencari data tidak mengalami kesusahan dalam mencari data yang diinginkan.

Mengisi tabel

Setelah tabel terbentuk, kita dapat memulai pekerjaan mengisi database. Syntax yang digunakan adalah :

mysql>insert into data (field1,field2) values (valuefield1,valuefield2);

contoh :

```
insert into data (nama, alamat) values ('firdaus', 'pustena');
```

perhatikan contoh di bawah ini :

```
mysql> insert into data (nama, alamat) values ('firdaus', 'pustena');
Query OK, 1 row affected (0.11 sec)
mysql>
```

yang perlu diperhatikan, untuk type field char data yang dimasukkan harus diapit tanda petik ' ', sedangkan untuk tipe integer tidak.

Untuk menampilkan isi dari input yang barusan kita isi gunakan sintak :

*mysql>select*from nama_table;*

contoh :

```
select*from data;
```

perhatikan gambar di bawah :

```
mysql> select * from data;
+-----+-----+
| nama  | alamat |
+-----+-----+
| firdaus | pustena |
+-----+-----+
1 row in set (1.10 sec)

mysql>
```

Tanda (*) merupakan perintah untuk menampilkan seluruh kolom beserta isinya. Jika kita ingin menampilkan hanya kolom nama saja, maka sintaknya adalah: [*mysql>select field from nama_table;*](#)

contoh :

```
select nama from data;
```

Perhatikan gambar di bawah :

```
mysql> select nama from data;
+-----+
| nama  |
+-----+
| firdaus |
+-----+
1 row in set (0.21 sec)

mysql>
```


SINTAK MySQL (Lanjutan)

- Editing Table
- Menambah/menghapus/mengubah field dari tabel
- Menampilkan isi tabel

Editing Table

Mengubah isi tabel

Untuk mengoreksi kesalahan data yang kita masukkan, atau untuk memperbaharui value field yang lama dapat kita gunakan syntax berikut;

[mysql>update nama_table set fieldl='valuefieldbaru';](#)

contoh :

update data set nama='adnan';

maka hasilnya adalah seluruh value field dari field yang ditentukan dari sintak di atas yang telah terisi maupun yang belum terisi akan berubah menjadi adnan, hal ini di sebabkan kita belum mengidentifikasi value field mana saja yang harus berubah.

Perhatikan gambar di bawah :

```
mysql> select*from data;
+-----+-----+
| nama      | alamat |
+-----+-----+
| firdaus   | pustena |
| alikomeini | salman |
+-----+-----+
2 rows in set (0.33 sec)

mysql> update data set nama='adnan';
ERROR 1062: Duplicate entry 'adnan' for key 1
mysql>
```

Kalimat error seperti di atas menandakan kalau ada kesalahan pengisian. Sebelumnya kita harus mengetahui kondisi field ketika pertama kali kita definisikan. Karena field nama merupakan *primary key* dari table tersebut maka tidak boleh value field-nya yang sama agar ketika user mencari data, tidak ada isi

data yang sama. Jadi sifatnya harus unik. Untuk itu kita harus mengupdate field selain field yang sudah kita definisikan sebagai primary key. Field lainnya adalah alamat.

Contoh:

update data set alamat='asrama';

perhatikan gambar di bawah :

```
mysql> update data set alamat='asrama';
Query OK, 2 rows affected (2.30 sec)
Rows matched: 2  Changed: 2  Warnings: 0

mysql> select * from data;
+-----+-----+
| nama      | alamat |
+-----+-----+
| adnan     | asrama |
| alikomeini | asrama |
+-----+-----+
2 rows in set (0.05 sec)

mysql>
mysql>
```

memperbaharui value field yang lama atau yang masih kosong tanpa harus merubah seluruh value field dalam satu field dapat kita gunakan syntax berikut;

[mysql>update nama_table set field='newvaluefield' where field2='valuefield2';](#)

contoh :

update data set alamat='pustena' where nama='adnan' ;

Maka yang berubah hanya isi kolom alamat dengan posisi sejajar dengan valuefield (firdaus) dari kolom nama.

Perhatikan gambar di bawah ini :

```
mysql> update data set alamat='pustena' where nama='adnan';
Query OK, 1 row affected (0.06 sec)
Rows matched: 1  Changed: 1  Warnings: 0

mysql> select * from data;
+-----+-----+
| nama      | alamat |
+-----+-----+
| adnan     | pustena |
| alikomeini | asrama |
+-----+-----+
2 rows in set (0.06 sec)

mysql>
```

Menghapus Isi Table

Untuk menghapus valuefield pada suatu tabel dapat dipakai syntax;

[mysql>delete from nama_table where field=valuefield;](#)

contoh :

```
delete from data where nama='alikoemini';
```

Hasilnya adalah seluruh valuefield yang satu baris dengan valuefield yang kita definisikan untuk di hapus akan terikut hilang.

Perhatikan gambar di bawah ini :

```
mysql> delete from data where nama='alikoemini';
Query OK, 1 row affected (1.32 sec)

mysql> select*from data;
+-----+-----+
| nama | alamat |
+-----+-----+
| adnan | pustena |
+-----+-----+
1 row in set (0.16 sec)

mysql>
```

Jika kita ingin menghapus valuefield tertentu saja tanpa harus menghapus valuefield lainnya maka gunakan sintak update seperti di atas namun pada bagian *valuefieldbaru* harus di kosongkan agar hasilnya nanti tidak ada data yang masuk sebagai input.

contoh :

```
update data set alamat='' where nama='adnan' ;
```

perhatikan gambar di bawah :

```
mysql> update data set alamat='' where nama='adnan';
Query OK, 1 row affected (0.50 sec)
Rows matched: 1  Changed: 1  Warnings: 0

mysql> select*from data;
+-----+-----+
| nama | alamat |
+-----+-----+
| adnan |         |
+-----+-----+
1 row in set (0.00 sec)

mysql>
```

Menambah/menghapus/mengubah field dari tabel

Bila kita ingin memodifikasi field pada tabel dapat digunakan syntax;

```
mysql>alter table nama_table add/drop field (typefield());
```

- Menambah field dalam tabel, sintaknya:

```
mysql>alter table nama_table add field typefield();
```

contoh:

```
alter table data add telepon int(10);
```

Perhatikan contoh di bawah :

```
mysql> alter table data add telepon int(10);
Query OK, 1 row affected (3.13 sec)
Records: 1 Duplicates: 0 Warnings: 0

mysql> select * from data;
+-----+-----+-----+
| nama  | alamat | telepon |
+-----+-----+-----+
| adnan |        | NULL    |
+-----+-----+-----+
1 row in set (0.16 sec)

mysql>
```

Pada gambar di atas ada tulisan NULL. Tulisan ini muncul karena ketika kita menulis sintaknya tidak kita sertai dengan kata not null seperti pada contoh di atas. Karena itu kata NULL akan otomatis muncul. Untuk menggantinya maka kita harus mengupdatenya dengan sintak seperti di atas.

- Menambah field dalam suatu table dengan mengatur posisi letak field, sintaknya

mysql>alter table nama_table add field typefield() after field;

contoh :

```
alter table data add pekerjaan char(15) after alamat;
```

Perhatikan gambar berikut :

```
mysql> alter table data add pekerjaan char after alamat;
Query OK, 1 row affected (1.43 sec)
Records: 1 Duplicates: 0 Warnings: 0

mysql> select * from data;
+-----+-----+-----+-----+
| nama  | alamat | pekerjaan | telepon |
+-----+-----+-----+-----+
| adnan |        | NULL      | NULL    |
+-----+-----+-----+-----+
1 row in set (0.16 sec)

mysql>
```

- Menghapus field dalam tabel, sintaknya :

mysql>alter table nama_table drop field;

contoh :

```
alter table data drop telepon;
```

Perhatikan gambar berikut :

```
mysql> alter table data drop telepon;
Query OK, 1 row affected (0.11 sec)
Records: 1 Duplicates: 0 Warnings: 0

mysql> select*from data;
+-----+-----+-----+
| nama | alamat | pekerjaan |
+-----+-----+-----+
| adnan |      | NULL      |
+-----+-----+-----+
1 row in set (0.05 sec)

mysql>
```

- Mengganti nama dari field table

Untuk mengganti penamaan judul field dari sebuah table gunakan sintak :

*mysql>alter table nama_table change name_field name_field_baru typefield
();*

contoh :

alter table data change pekerjaan telepon int(10);

perhatikan gambar berikut :

```
mysql> alter table data change pekerjaan telepon int(10);
Query OK, 1 row affected (1.37 sec)
Records: 1 Duplicates: 0 Warnings: 0

mysql> select*from data;
+-----+-----+-----+
| nama | alamat | telepon |
+-----+-----+-----+
| adnan |      | NULL    |
+-----+-----+-----+
1 row in set (0.11 sec)

mysql>
```

Perhatikan gambar tersebut. Sebelumnya kita telah mendefinisikan typefield pekerjaan adalah “char”, tetapi dengan sintak di atas kita dapat sekaligus mengganti tipefield dari field yang ingin kita ganti namanya dan juga kita dapat memperbesar atau memperkecil ukuran tipefieldnya. Bila kita tidak ingin mengganti nama fieldnya tetapi hanya ingin memperbesar ukuran tipefieldnya sama maka sintaknya sama saja seperti di atas tetapi pada kalimat *nama_field_baru* tetap di tulis *nama_field* yang awal sedangkan ukurannya dapat kita ganti di dalam kurung ().

Menampilkan isi tabel

untuk menampilkan keseluruhan isi table telah kita bahas di atas yaitu dengan menggunakan syntax

*[mysql>select * from nama_field;](#)*

untuk menampilkan tabel dengan kondisi field tertentu semisal field nama

[mysql>select field1,field2 from nama_field;](#)

contoh :

select nama, alamat from data;

Menampilkan isi tabel dengan beberapa keyword :

1. Distinct

Keyword ini berguna untuk menghilangkan record-record yang sama.

Sintaknya :

[mysql>select distinct field from nama_table ;](#)

contoh :

select distinct nama from data;

2. Between

Keyword ini berguna untuk membatasi suatu kolom yang berada pada suatu batas nilai tertentu. Misalkan, buka tabel user di database formulir, sintaknya :

[mysql>select field1, field2 from nama_table where field2 between valuefield2 and valuefield2;](#)

contoh :

select nama, umur from user where telepon between 2500000 and 2530000;

3. like

Keyword ini berguna untuk mencari data yang memiliki pola tertentu. Misalkan untuk mencari alamat user dengan kata unix, dapat digunakan query :

[mysql>select field from nama_table where field like '%value%';](#)

contoh :

```
select alamat from data where alamat like '%cisitu%';
```

4. order by

Keyword ini berguna untuk menampilkan field sesuai abjad. Sintak

:

```
mysql>select * from nama_table order by field;
```

contoh :

```
select*from data order by nama;
```

untuk urutan kebalikannya dapat digunakan;

```
mysql>select field1,field2 from data order by field1 desc;
```

Untuk sintak-sintak di atas gunakan sample data seperti yang ada di dalam gambar berikut:

```
mysql> select*from data;
+-----+-----+-----+
| nama   | alamat          | telepon |
+-----+-----+-----+
| adnan  | bgfamily        | 250000  |
| firdaus| asrama bg       | 2500114 |
| faisal | kmrg-itb        | 2530115 |
| razi   | cisitu indah 14| 2503508 |
| boim   | bgfamily        | 2547890 |
+-----+-----+-----+
5 rows in set (0.00 sec)

mysql>
```

AGREGATE FUNCTION DAN PHPMYADMIN

- Agregate Function
- Sub Agregate Dengan Group By
- Menyaring Hasil Agregate Function

PHPMYADMIN

- Instalasi
- Cara Kerja PhpMyadmin

Agregate Function

Selain dari pengolahan record-record, SQL juga menyediakan fungsi-fungsi agregate dalam SQL adalah untuk menghitung hasil tampilan. Karena ia sifatnya adalah untuk di tampilkan maka fungsi tersebut tergolong dalam bagian sintak *select*. Beberapa jenis fungsinya antara lain :

Nama fungsi	Fungsi
sum()	Menghitung jumlah ekspresi numerik
avg()	Menghitung rata-rata ekspresi numerik
min()	Menghitung angka minimal ekspresi numerik
max()	Menghitung angka maksimal ekspresi numerik
count()	Menghitung jumlah non-null ekspresi
count(*)	Menghitung jumlah baris

Untuk lebih jelas, perhatikan gambar berikut :

```
mysql> select*from data;
+-----+-----+-----+-----+
| nama   | alamat          | telepon | umur |
+-----+-----+-----+-----+
| adnan  | bgfamily        | 250000  | 19   |
| firdaus| asrama bg       | 2500114 | 20   |
| faisal | kmrg-itb        | 2530115 | 22   |
| razi   | cisitu indah 14 | 2503508 | 20   |
| boim   | bgfamily        | 2547890 | 21   |
+-----+-----+-----+-----+
5 rows in set (0.27 sec)
mysql> _
```

Dari table di atas, perhatikan pada kolom telepon dan umur, karena kedua kolom tersebut yang memiliki value aritmatika.

Perhatikan sintak dasar dari function aggregate di bawah ini :

```
mysql> select aggregate_function(nama_field) from nama_table;
```

contoh 1:

```
select sum(umur) from data;
```

Perhatikan hasilnya dalam gambar berikut :

```
mysql> select sum(umur) from data;
+-----+
| sum(umur) |
+-----+
|         102 |
+-----+
1 row in set (1.54 sec)
mysql>
```

Dari gambar di atas jelas bahwa fungsi dari sum adalah menjumlahkan seluruh data yang bersifat arimatika.. Sedangkan kalau data yang ada di dalam kolom tersebut tidak bersifat arimatika (angka) maka hasilnya adalah nol. Sedangkan bila dalam satu kolom terdapat berbagai jenis data seperti data angka maupun data kata/kalimat maka data yang terbaca adalah data aritmatika dan data non angka di anggap tidak ada atau nol.

Contoh 2:

```
Select avg(umur) from data;
```

Maka hasilnya adalah :

```
mysql> select avg(umur) from data;
+-----+
| avg(umur) |
+-----+
|    20.4000 |
+-----+
1 row in set (0.16 sec)
mysql>
```

Seluruh data di jumlahkan kemudian di bagi dengan banyak data, sehingga menghasilkan data seperti di atas.

Sub Agregate Dengan Group By

Fungsi Group By adalah untuk menyatukan antara beberapa field dalam sebuah table di mana salah satu field atau lebih menggunakan agregate function. Misalnya, jika kita ingin menampilkan nama dengan rata-rata umur setiap pelajar maka kita harus menggunakan sintak :

```
mysql>select field1, aggregate_function(field2) from nama_table order by field1;
```

contoh :

```
select alamat, avg(umur) from data group by alamat;
```

Maka hasilnya ada di gambar bawah ini :

```
mysql> select alamat, avg(umur) from data group by alamat;
+-----+-----+
| alamat | avg(umur) |
+-----+-----+
| asrama bg | 20.0000 |
| bgfamily | 20.0000 |
| cisitu indah 14 | 20.0000 |
| kmrg-itb | 22.0000 |
+-----+-----+
4 rows in set (0.00 sec)

mysql>
```

Menyaring Hasil Agregate Function

Untuk menyaring tampilan setelah dilakukan fungsi agregate, maka harus menggunakan keyword *having*. Misalnya untuk menyaring pelajar dengan umur di bawah 19 tahun, maka sintaknya adalah :

```
mysql>select field1, aggregate_function(field2) from nama_table group by field1 having agragate_function(field2) operator_perbandingan valuefield;
```

contoh :

```
select alamat, avg (umur) from data group by alamat having avg(umur) >20;
```

Maka hasilnya adalah :

```
mysql> select alamat, avg(umur) from data group by alamat having avg(umur)>20;
+-----+-----+
| alamat | avg(umur) |
+-----+-----+
| kmrg-itb | 22.0000 |
+-----+-----+
1 row in set (0.11 sec)

mysql>
```

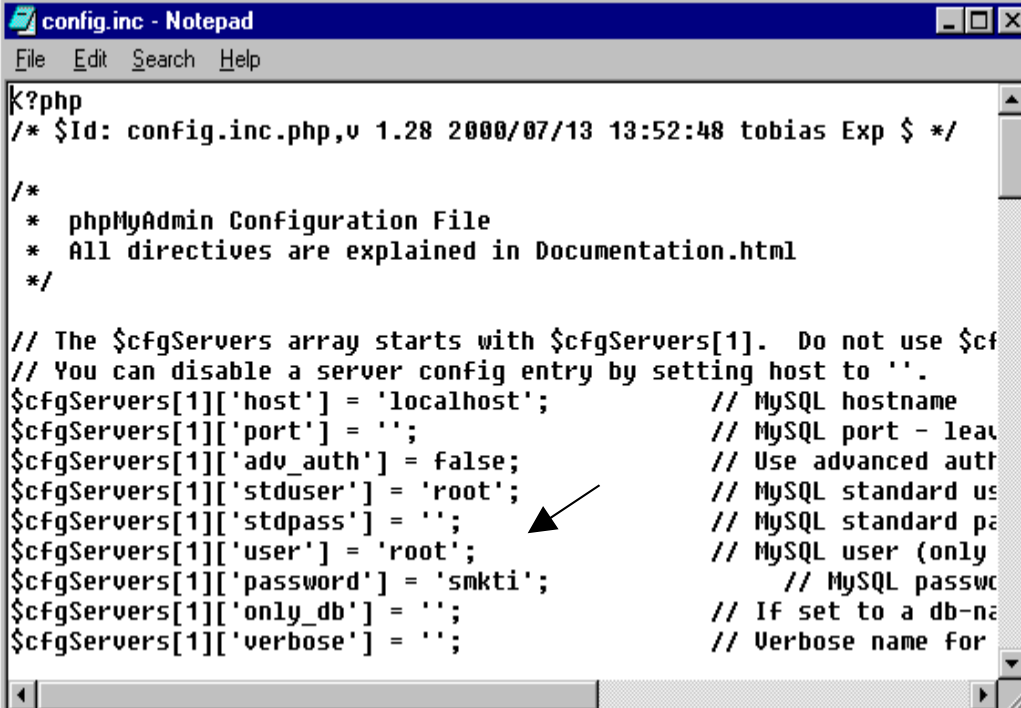
PHPMYADMIN

Instalasi

Phpmysql merupakan software yang digunakan untuk memudahkan bagi seseorang untuk membuat suatu database dengan cepat tanpa harus menetik secara manual di Dos-Prompt. Software ini dapat di download di <http://www.phpwizard.net>.

Setelah di download dan di ekstrak maka hasilnya letakkan di folder di mana kita posisikan document Root web server kita. Peletakan ini akan kita pelajari pada modul PHP yang akan datang. Namun sebelumnya ada file yang perlu di edit, yaitu file yang bernama *config.inc.php3*. Di dalam file ini ada beberapa kalimat yang harus di edit yaitu

Pada bagian user dan password. Pada kedua baris itu kita harus menyamakannya dengan kondisi ketika kita pertama kali menginstal MySQL melalui Dos_Prompt. Jika user-nya root maka tulis di bagian tersebut root, dan jika password MySQL-nya smkti maka letakkan kata smkti di baris tersebut. Untuk lebih jelasnya lagi perhatikan gambar di bawah ini :



```
config.inc - Notepad
File Edit Search Help
k?php
/* $Id: config.inc.php,v 1.28 2000/07/13 13:52:48 tobias Exp $ */

/*
 * phpMyAdmin Configuration File
 * All directives are explained in Documentation.html
 */

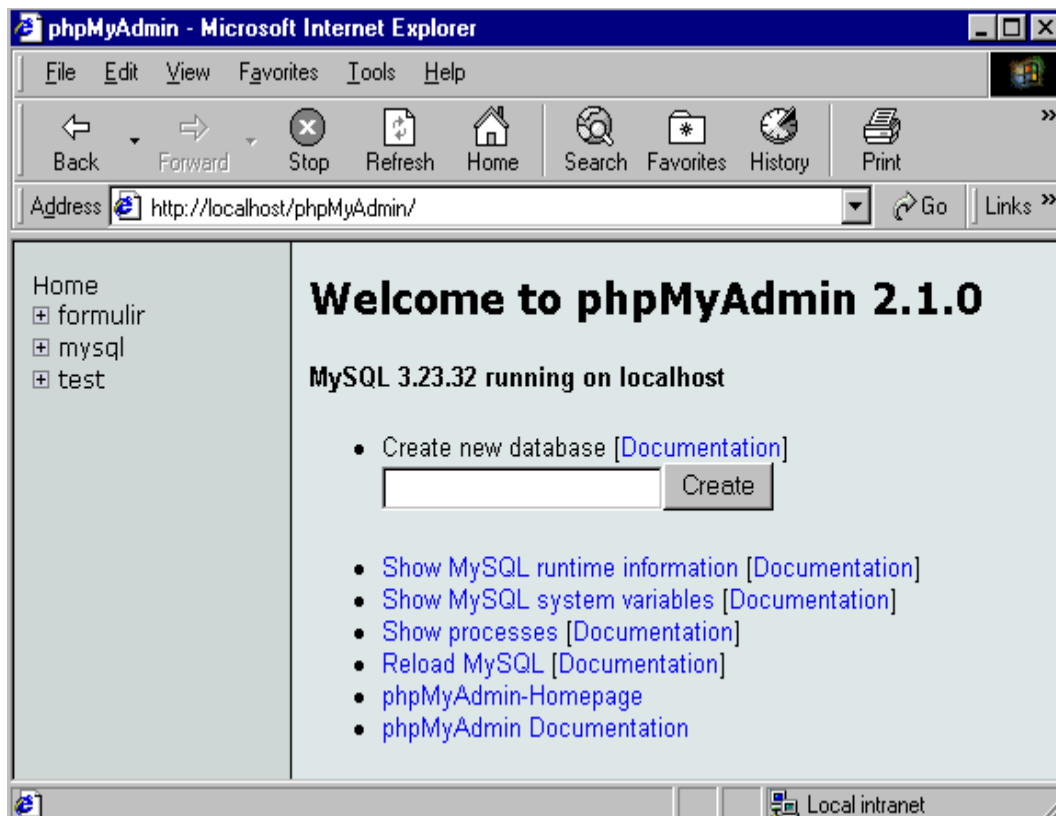
// The $cfgServers array starts with $cfgServers[1]. Do not use $cf
// You can disable a server config entry by setting host to ''.
$cfgServers[1]['host'] = 'localhost';           // MySQL hostname
$cfgServers[1]['port'] = '';                    // MySQL port - lea
$cfgServers[1]['adv_auth'] = false;             // Use advanced aut
$cfgServers[1]['stduser'] = 'root';            // MySQL standard us
$cfgServers[1]['stdpass'] = '';                // MySQL standard pa
$cfgServers[1]['user'] = 'root';               // MySQL user (only
$cfgServers[1]['password'] = 'smkti';         // MySQL passwo
$cfgServers[1]['only_db'] = '';               // If set to a db-na
$cfgServers[1]['verbose'] = '';              // Uerbose name for
```

Setelah selesai mengeditnya kemudian muka webserver yang telah ada yaitu apache kemudian buka browser IE atau netscape, tulis di addressnya dengan localhost, maka secara otomatis browser akan membuka directory document root yang telah kita buat sebelumnya. Untuk lebih jelasnya lagi mengenai pembuatan web server dengan apache dan menjalankan server-side scripting yaitu PHP, akan kita bahas pada modul berikutnya.

Cara Kerja PhpMyadmin

Setelah berhasil masuk kedalam localhost maka kita masuk kedalam folder PhpMyadmin yang telah kita letakkan sebelumnya. Kemudian klik dua kali file *index.php3* yang digunakan untuk menjalankan software tersebut. Ada cara lain tanpa harus mengklik file *index.php3* yaitu mengedit file *httpd.conf* pada bagian “Directory Index” kita harus menulis ekstention *.php3* atau *.php* agar ketika browser membuka folder tersebut, file tersebut otomatis terbuka tanpa harus mengklik file tersebut.

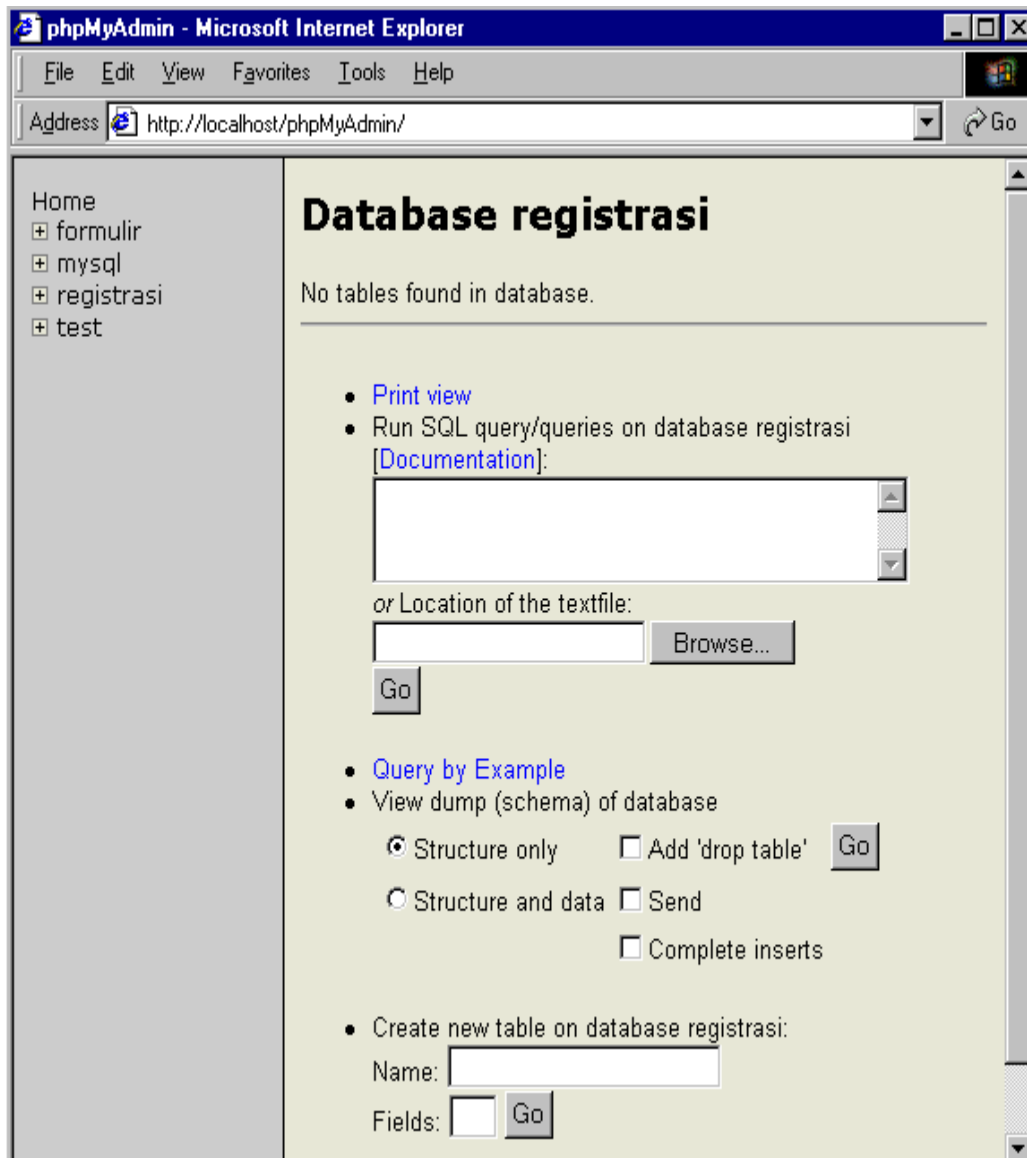
Setelah software tersebut bekerja maka akan tampil browser seperti gambar di bawah ini :



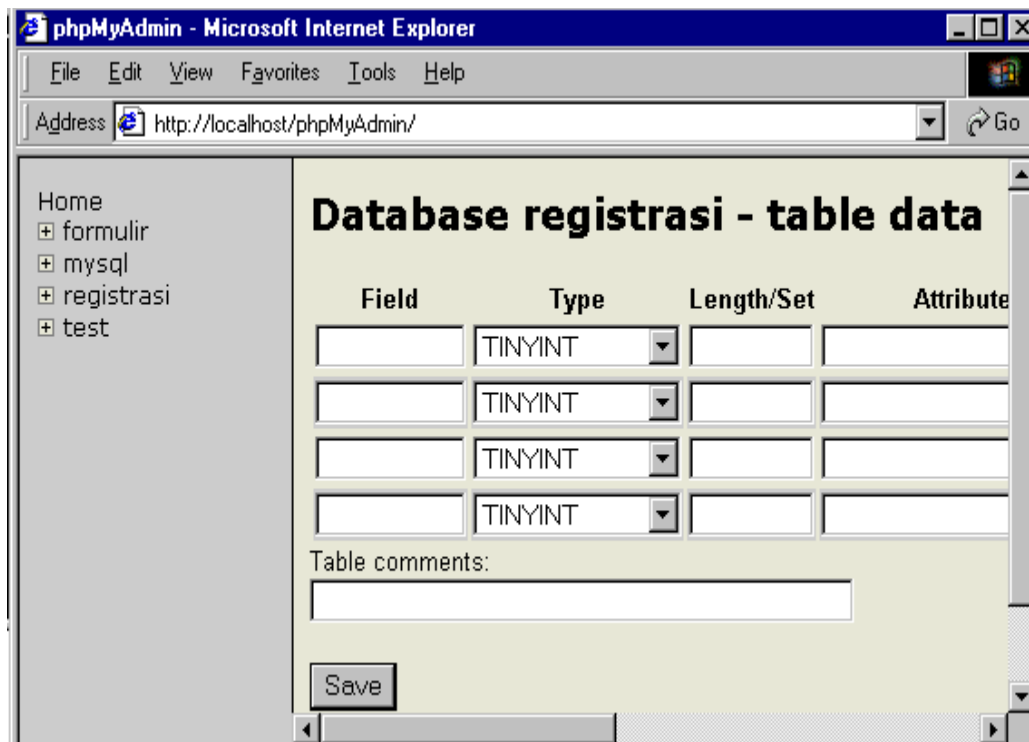
Pada browser tersebut tampak beberapa bagian yang menyatakan nama database yang telah kita buat sebelumnya dengan menggunakan Dos-Prompt yaitu di sebelah kiri gambar sedangkan di sebelah kanan tampak suatu kolom yang di peruntukkan untuk membuat suatu database yang baru. Kalau kita tidak ingin membuat database yang baru dan hanya ingin mengedit database yang lama maka anda cukup mengklik nama database yang telah ada di sebelah kiri tersebut, maka Phpmysql akan membawa anda ke halaman lainnya yang berupa tempat editing database lengkap dengan field dan valuefieldnya.

Dalam modul ini kita akan membuat suatu database yang baru dengan nama database registrasi. Untuk itu kita tinggal menuliskan kata registrasi di kolom yang telah tersedia kemudian klik button *create*.

Maka akan tampil suatu tampilan sebagai berikut :

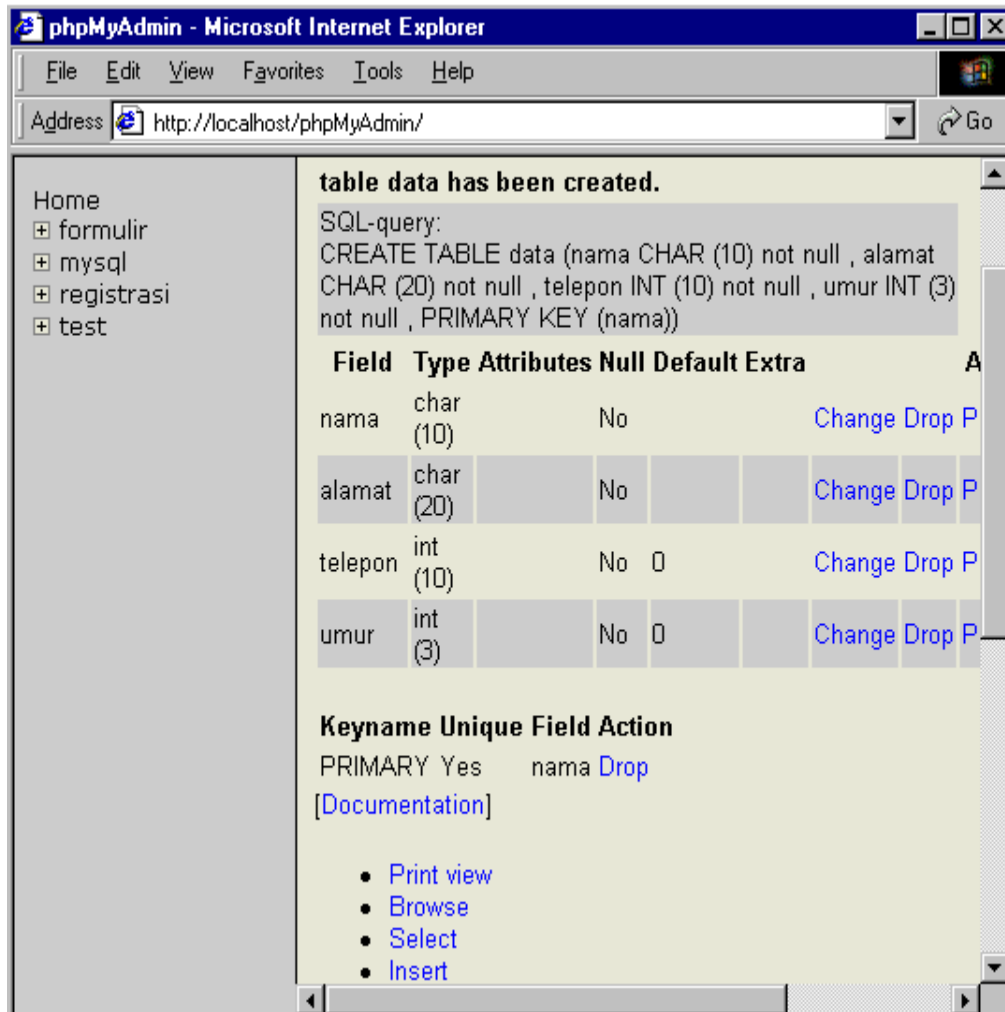


Pada bagian paling bawah yaitu *create new table*, di bawahnya terdapat field name dan fields. Kedua kolom tersebut adalah digunakan ketika kita ingin membuat table baru di dalam database tersebut. Misalkan nama tablenya adalah data dan banyak fields-nya adalah 4, kemudian tekan button “go”. Maka akan tampak tampilan sebagai berikut :



Pada gambar di atas, banyak fieldnya kurang lengkap, hal ini karena keterbatasan kertas. Untuk lebih jelasnya lagi, anda dapat melihatnya sendiri jika telah berhasil menjalankannya. Di dalam gambar tersebut anda tinggal memasukkan nama field, typefield, besar tipefieldnya dan apakah kita posisikan field tersebut sebagai primary key atau tidak. Untuk kondisi primary key di gambar atas tidak kelihatan. Posisinya ada di sebelah paling kanan dari gambar.

Setelah semuanya di isi, maka tekan button save, maka anda akan di tampilkan suatu halaman lainnya yaitu :



Pada gambar di atas, tampak pada bagian di atas adalah sintak SQL yang jika kita tulis di tampilan Dos_Prompt akan sama hasilnya yaitu akan menghasilkan suatu table dengan nama data dan nama fieldnya adalah *nama*, *alamat*, *telepon*, dan *umur*. Pada halaman ini kita dengan mudah memasukkan data ke table yaitu dengan menekan dua kali kata insert, maka kita akan di tampilkan suatu field-field yang memiliki nama seperti yang kita buat sebelumnya. Untuk melihat seluruh hasil input ke database, kita cukup mengklik kata browse, maka kita akan di tampilkan seluruh isi field yang telah ada.