**MODUL PRAKTIKUM BASIS DATA**

**MODUL 1 – PENGENALAN MYSQL**

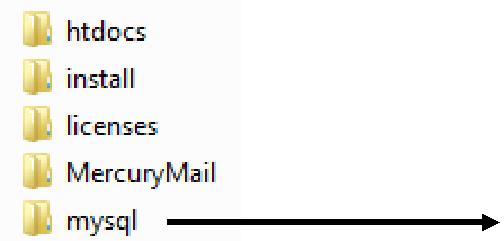
MySQL merupakan salah satu DBMS yang bersifat freeware, sehingga menjadi popular dikalangan pengguna database. Selain karena Free MySQL juga merupakan DBMS yang relatif ringan dan mudah digunakan. Ada beberapa software yang bisa digunakan untuk menjalankan service MySQL, diantaranya WAMP, PhpTriad dan XAMPP. Namun XAMPP merupakan yang paling popular saat ini karena masih terus dilakukan update pada software nya.

XAMPP merupakan program paket yang didalamnya terdapat beberapa software yang digabungkan menjadi satu. Dalam XAMPP sendiri didalamnya terdapat 5 software yang berbeda (MySql, Apache, FileZilla,Mercury, Tomcat). Jadi ketika menginstall XAMPP kita tidak perlu lagi menginstall Apache dan MySQL secara terpisah.

1. Install MySQL (XAMPP)

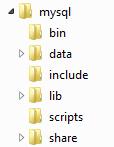
Untuk master software XAMPP bisa didownload gratis di situ resminya www.apachefriends.org/en/**xampp**.html *.* Hal yang perlu diingat ketika menginstall XAMPP adalah direktori tempat meletakan hasil installan.

1. Direktori MySQL



Direktori mysql

Semua proses yang berjalan untuk bekerja dalam MySQL terdapat dalam direktori mysql. Ada beberapa direktori yang perlu diketahui dalam direltori mysql :



a. Bin

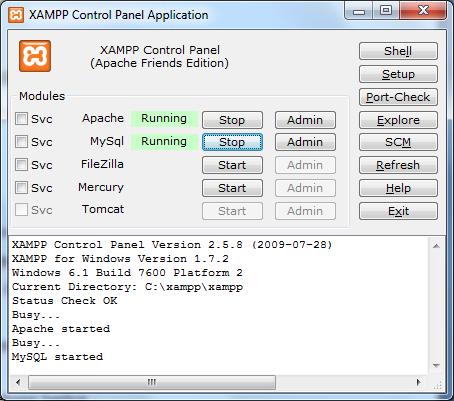
Direktori bin berisi service-service yang bisa dijalankan untuk mengakses mysql.

* 1. Data

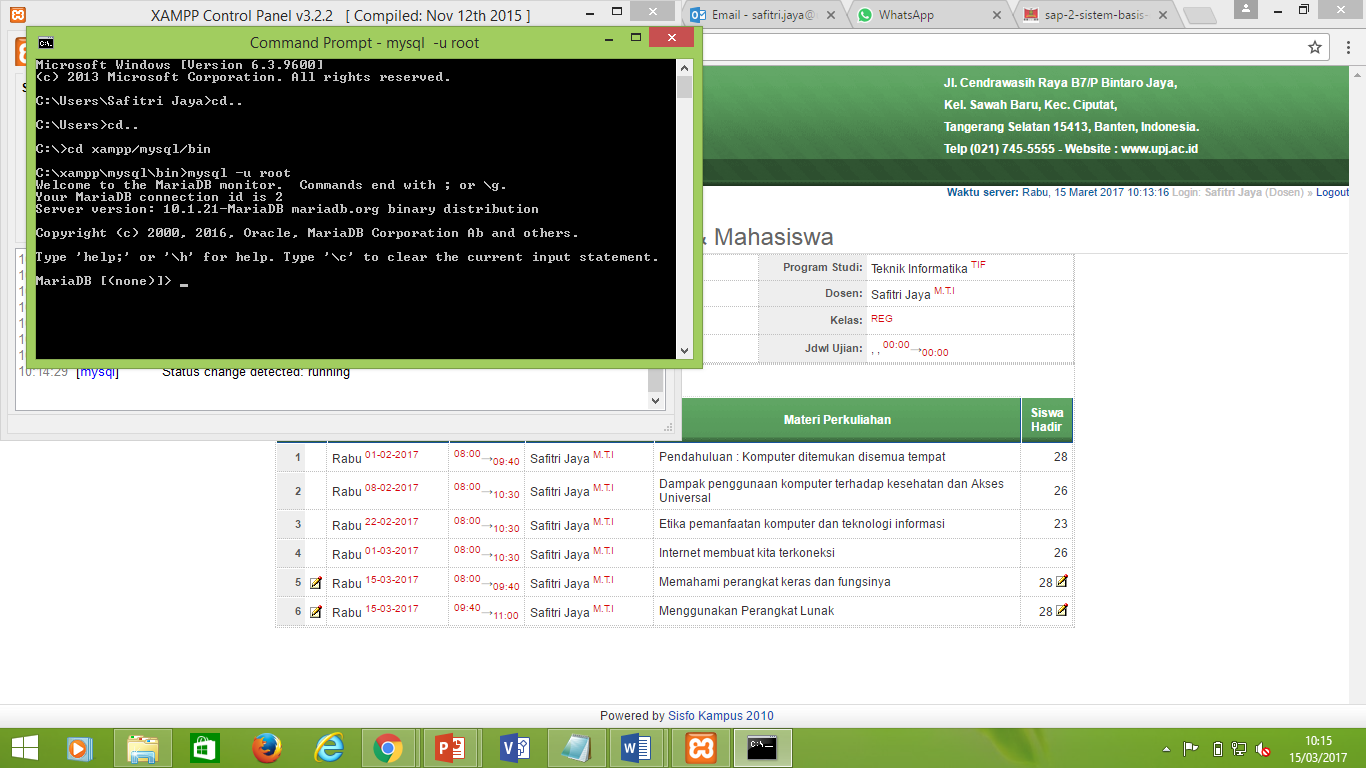
Direktori ini berisi database dan tabel yang sudah dibuat.

1. Menggunakan MySQL

a. Nyalakan (START) Service MySQL



1. Masuk kedalam MySQL melalui CMD dengan mengetikkan perintah sebagai berikut :



**MODUL 2 – SHOW, USE, CREATE and DESC**

* 1. **SHOW**

Berfungsi untuk melihat list/daftar dari database atau tabel yang sudah dibuat.

1. Melihat daftar database

Untuk melihat daftar database yang sudah dibuat. Query : **SHOW** databases;

1. Melihat daftar tabel

Untuk melihat daftar database yang ada digunakan fungsi show tables. Query : **SHOW** tables;

* 1. **USE**

Perintah USE digunakan untuk memilih database yang ingin digunakan. Query : **USE** nama\_database;

* 1. **CREATE**

Berfungsi untuk membuat database atau tabel baru.

* 1. Membuat database

**CREATE DATABASE** nama\_database;

b. Membuat tabel

**CREATE TABLE** nama\_tabel (

Field1 TipeData1,

Field2 TipeData2,

………………

FieldN TipeDataN );

**2.4 DESC**

Digunakan untuk melihat struktur (metadata) sebuah tabel.

**DESC** nama\_tabel;

**Latihan**

1. Buat sebuah database mahasiswa.

CREATE DATABASE mahasiswa;

2. Buat dua buah tabel siswa dan dosen di dalam database mahasiswa

CREATE TABLE siswa (

Nim int not null,

Nama char(20),

Wali int );

CREATE TABLE wali (

Nip int not null,

Nama char(20),

Alamat char(50) );

**Tugas**

1. Buat satu buah database
2. Buat dua buah tabel pada database yang telah dibuat sebelumnya

**MODUL 3 – INSERT, SELECT, WHERE**

1. **Insert**

Berfungsi untuk mengisikan data kedalam sebuah tabel.

Query :

**INSERT** **INTO** nama\_tabel **VALUES** (‘nilaiField1’, ‘nilaiField2’, ………,

‘nilaiFieldN’);

1. **Select**

Digunakan untuk menampilkan isi tabel. Untuk menampilkan sebuah tabel ada beberapa kondisi yang bisa digunakan, antara lain :

* 1. SELECT ALL

Kondisi ini digunakan untuk menampilkan semua record dan semua field nya dalam sebuah tabel.

Query : **SELECT \* FROM** nama\_tabel**;**

* 1. SELECT FIELD

Kondisi ini digunakan untuk menampilkan remua record yang ada, tetapi hanya field tertentu saja.

Query : **SELECT** field1,field2,… **FROM** nama\_tabel**;**

* 1. SELECT RECORD (WHERE)

Kondisi ini digunakan untuk menampilkan isi dari record tertentu saja. WHERE bisa digunakan untuk SELECT ALL maupun SELECT FIELD.

Query : **SELECT (ALL/FIELD) FROM** nama\_tabel **WHERE** nama\_field = ‘value’**;**

**Latihan**

1. Isikan data kedalam tabel siswa dan dosen yang telah dibuat sebelumnya

a. Mengisi data pada tabel siswa

INSERT INTO siswa VALUES (‘123040203’, ‘WAWAN ADI’, ‘19721201’);

INSERT INTO siswa VALUES (‘123040204’, ‘INDRA KUSUMA’, ‘19721204’);

INSERT INTO siswa VALUES (‘123040205’, ‘DIANA FITRIA’, ‘19721203’);

INSERT INTO siswa VALUES (‘123040206’, ‘BAYU PUTRA’, ‘19721201’);

INSERT INTO siswa VALUES (‘123040207’, ‘ADRIANTI’, ‘19721202’);

INSERT INTO siswa VALUES (‘123040208’, ‘AHMAD AFANDI’, ‘19721204’);

* 1. Mengisi data pada tabel wali

INSERT INTO wali VALUES (‘19721201’, ‘WAWAN ADI’, ‘19721201’); INSERT INTO walia VALUES (‘19721202’, ‘INDRA KUSUMA’, ‘19730412’); INSERT INTO wali VALUES (‘19721203’, ‘DIANA FITRIA’, ‘19721201’); INSERT INTO wali VALUES (‘19721204’, ‘BAYU PUTRA’, ‘19721201’);

2. Tampilkan isi data dari tabel siswa dan dosen

a. Menampilkan isi tabel siswa

SELECT \* FROM siswa;

b. Menampilkan isi tabel dosen

SELECT \* FROM dosen;

c. Menampilkan nip dan nama dari tabel wali

SELECT nip, nama FROM wali;

d. Menampilkan record dengan nim = 123040208 dari tabel siswa

SELECT \* FROM siswa WHERE nim = ‘123040208’;

e. Menampilkan nim dan wali dengan nim = 123040208 dari tabel siswa

SELECT nim, wali FROM siswa WHERE nim = ‘123040208’;

**Tugas**

1. Tampilkan record dengan nim = 123040204 atau wali = 19721204.
2. Isikan data pada tabel yang telah dibuat pada tugas sebelumnya.
3. Tampilkan isi data pada tabel yang telah dibuat pada tugas sebelumnya.

**MODUL 4 – KONDISI**

Kondisi yang dibentuk pada perintah-perintah SQL dapat dalam kriteria yang sifatnya pembandingan dengan menggunakan tanda ‘=’ untuk mencari suatu record maupun tidak. Beberapa kondisi dalam SQL :

1. **LIKE**

Menampilkan isi record dengan menggunakan sebagian dari isi sebuah data. Secara fungsi LIKE berfungsi sama seperti WHERE (=).

Query : **SELECT \* FROM** nama\_tabel **WHERE** nama\_field **LIKE ‘**kondisi**’;**

Ada terdapat 3 kondisi di dalam like :

* 1. %data%

Persen didepan dibelakang menunjukan bahwa record yang ingin ditampilkan adalah semua record yang mengandung “data” pada sebuah field. “data” bisa berada diawal, tengah ataupun akhir.

* 1. %data

Persen didepan menunjukan bahwa record yang ingin ditampilkan adalah semua record yang mengandung “data” pada sebuah field dan “data” harus berada diawal.

* 1. data%

Persen dibelakang menunjukan bahwa record yang ingin ditampilkan adalah semua record yang mengandung “data” pada sebuah field dan “data” harus berada akhir.

1. **ORDER BY**

Jika sebuat SELECT tidak memiliki ORDER BY, maka record akan ditampilkan sesuai dengan urutan penginputan datanya. ORDER BY memungkinkan menampilkan hasil SELECT sesuai dengan urutan yang diinginkan.

Query : **SELECT \* FROM** nama\_tabel **ORDER BY** nama\_field**;**

1. **ASC/DESC**

Dalam mengurutkan terdapat dua kondisi : Mengurutkan data dengan terurut naik (ASCENDING) atau terurut menurun (DESCENDING).

1. **GROUP BY**

GROUP BY memungkinkan mengelompokan hasil dari SELECT berdasarkan persamaan isi datanya.

Query : **SELECT \* FROM** nama\_tabel **ORDER BY** nama\_field**;**

**Latihan**

Database : penjualan

Tabel : pelanggan

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **field** | | |  |  | **type** |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | kode | | |  |  | Int(3) |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | nama | | |  |  | Char (25) |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | alamat | | |  |  | Char (35) |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | kota | | |  |  | Char (15) |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | notelp | | |  |  | Char (12) |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | tipe | | |  |  | Char (10) |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
|  | **kode** |  |  |  |  | **nama** |  |  |  |  | **alamat** |  |  | **kota** |  |  | **notelp** |  |  | **tipe** |  |
|  | 123 |  |  | PASTRIA SANDRA | | |  |  |  |  | JL KLEDOKAN NO 21 | |  | JOMBANG | | 08123456789 | |  |  | GOLD | |
|  | 124 |  |  | IHSAN YUSUF | | |  |  |  |  | JL KAPAS N0 3 | |  | BEKASI | | 08123456788 | |  |  | PREMIUM | |
|  | 125 |  |  | RIO DEWANTO | | |  |  |  |  | JL MERDEKA NO 14 | |  | JAKARTA | | 08123456787 | |  |  | PREMIUM | |
|  | 126 |  |  | SANDRA DEWI | | |  |  |  |  | JL PELANGI NO 2 | |  | JAKARTA | | 08123456786 | |  |  | PREMIUM | |
|  | 127 |  |  | DEWA PUTRA | | |  |  |  |  | JL MACAN NO 34 | |  | SURABAYA | | 08123456785 | |  |  | GOLD | |
|  | 128 |  |  | PUTRA SANTOSA | | |  |  |  |  | JL KELINCI NO 23 | |  | SURABAYA | | 08123456784 | |  |  | GOLD | |
|  | 129 |  |  | ELYN GAURA | | |  |  |  |  | JL MELATI NO 12 | |  | YOGYAKARTA | | 08123456783 | |  |  | ECONOMY | |
|  | 130 |  |  | LETISIA RURON | | |  |  |  |  | JL MERPATI NP 26 | |  | YOGYAKARTA | | 08123456782 | |  |  | ECONOMY | |
|  | 131 |  |  | KARTINI | | |  |  |  |  | JL DANAU TOBA NO 12 | |  | SEMARANG | | 08123456781 | |  |  | PREMIUM | |
|  | 132 |  |  | PUTRI DIANA | | |  |  |  |  | JL DANAU KERINCI NO 13 | |  | BEKASI | | 08123456799 | |  |  | GOLD | |
|  | 133 |  |  | JULIA RAHMAN | | |  |  |  |  | JL ELANG NO 32 | |  | JAKARTA | | 08123456798 | |  |  | ECONOMY | |
|  | 134 |  |  | PUTRA PERDANA | | |  |  |  |  | JL TULIP NO 12 | |  | JOMBANG | | 08123456797 | |  |  | PREMIUM | |
|  | 135 |  |  | PRANANDA GITA | | |  |  |  |  | JL SEDAP MALAM N0 27 | |  | SEMARANG | | 08123456796 | |  |  | ECONOMY | |
|  | 136 |  |  | SATRIA ARDI | | |  |  |  |  | JL SRIWIJAYA NO 3 | |  | MATARAM | | 08123456795 | |  |  | GOLD | |
|  | 137 |  |  | ANANDA DILA | | |  |  |  |  | JL KAWI NO 12 | |  | SURABAYA | | 08123456794 | |  |  | GOLD | |
|  | 138 |  |  | ALDILA BAGUS | | |  |  |  |  | JL AFFANDI N0 23 | |  | YOGYAKARTA | | 08123456793 | |  |  | PREMIUM | |

1. Mengurutkan berdasarkan nama DESC

Query : SELECT \* FROM pelanggan ORDER BY nama DESC;

1. Mengelompokan record berdasarkan tipe

Query : SELECT \* FROM pelanggan GROUP BY tipe;

1. Menampilkan record dengan memiliki data pada field nama = ‘dewa’

Query : SELECT \* FROM pelanggan WHERE nama LIKE ‘%dewa%’;

**Tugas**

Gunakan fungsi LIKE :

1. Tampilkan record dengan kode 123 & 136. Urutkan berdasarkan alamat asc
2. Tampilkan record dengan kode 128 & 134.
3. Tampilkan record dengan nomor rumah 12, urutkan berdasarkan tipe desc.
4. Tampilkan record dengan kode 132, 134, 135, 137. Urutkan berdaskan nama asc.
5. Tampilkan record dengan kode 127, 128, 134. Urutkan berdasarkan alamat.



**MODUL 5 - UPDATE, DELETE, ALTER, DROP, DESC**

1. **UPDATE**

Fungsi UPDATE digunakan untuk merubah isi data field dari sebuah record.

Query :

**UPDATE** nama\_tabel **SET** nama\_field **= ‘**value**’ WHERE** nama\_field **= ‘**value**’**

1. **DELETE**

Fungsi DELETE digunakan untuk menghapus sebuah record dalam sebuah database.

Query : **DELETE** nama\_tabel **WHERE** nama\_field **= ‘**value**’**

1. **ALTER**

Perintah ALTER berfungsi untuk mengubah struktur dari sebuah tabel. Bisa berarti menambahkan field baru, merubah field yang sudah ada, maupun menghapus field yang sudah ada.

Query :

* **ALTER TABLE** nama\_tabel **ADD** new\_field tipedata
* **ALTER TABLE** nama\_tabel **CHANGE** field\_lama field\_baru tipedata
* **ALTER TABLE** nama\_tabel **DROP** nama\_field

1. **DROP**

Perintah DROP digunakan untuk menghapus struktur dalam database, termasuk menghapus database dan tabel.

Query :

* **DROP DATABASE** nama\_database**;**
* **DROP TABLE** nama\_tabel**;**

1. **DESC**

Perintah DESC digunakan untuk melihat struktur/metadata dari sebuah tabel.

Query : **DESC** nama\_table**;**

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | **Latihan** | | | |  |  |  |  |  |  |
|  | Database : penjualan | | | |  |  |  |  |  |  |
|  | Tabel : barang | | | |  |  |  |  |  |  |
|  |  |  |  |  |  |  |  |  |  |  |
|  |  | **Field** | |  | **Tipedata** | |  |  |  |  |
|  | ID |  |  |  | CHAR (PK) | |  |  |  |  |
|  | NAMA\_BARANG | | |  | CHAR | |  |  |  |  |
|  | JENIS\_BARANG | | |  | CHAR | |  |  |  |  |
|  | HARGA\_BELI | | |  | INT | |  |  |  |  |
|  | HARGA\_JUAL | | |  | INT | |  |  |  |  |
|  | STOK |  |  |  | INT | |  |  |  |  |
|  | Isi tabel BARANG | | | |  |  |  |  |  |  |
|  |  |  |  | | |  | |  |  |  |
|  | **ID** |  | **NAMA\_BARANG** | | | **JENIS\_BARANG** | | **HARGA\_BELI** | **HARGA\_JUAL** | **STOK** |
|  | B01 |  | RINSO | |  | DETERGEN | | 10000 | 11000 | 10 |
|  | B02 |  | DAIA | |  | DETERGEN | | 9000 | 10000 | 20 |
|  | B03 |  | LIFEBUOY | |  | SABUN MANDI | | 2000 | 3000 | 30 |
|  | B04 |  | LIFEBUOY | |  | SHAMPOO | | 12000 | 14000 | 20 |
|  | B05 |  | LUX | |  | SABUN MANDI | | 2000 | 3000 | 12 |
|  | B06 |  | DETTOL | |  | SABUN MANDI | | 3000 | 4000 | 15 |
|  | B07 |  | CITRA | |  | BODY LOTION | | 12000 | 15000 | 16 |
|  | B08 |  | CITRA | |  | SABUN MANDI | | 2000 | 2500 | 25 |
|  | B09 |  | PANTENE | |  | SHAMPOO | | 16000 | 17000 | 30 |
|  | B10 |  | PEPSODENT | |  | PASTA GIGI | | 9000 | 11000 | 19 |

1. Mengubah stok baris pertama menjadi 15

**UPDATE** barang **SET** stok = ‘15’ **WHERE** id = ‘B01’;

1. Menghapus record pada baris pertama

**DELETE FROM** barang **WHERE** id = ‘B01’;

1. Menambahkan sebuah kolom baru dengan nama “DISTRIBUTOR” bertipe CHAR

**ALTER TABLE** barang **ADD** distributor char(15);

1. Mengubah nama kolom DISTRIBUTOR menjadi PENYALUR bertipe CHAR

**ALTER TABLE** barang **CHANGE** distributor penyalur char(15);

1. Menghapus kolom PENYALUR

**ALTER TABLE** barang **DROP** penyalur;

**SOAL**

1. Ubah harga beli pada baris terakhir menjadi 10000
2. Ubah harga jual untuk LIFEBUOY jenis SABUN MANDI menjadi 4000
3. Ubah harga jual dan harga beli untuk SHAMPO PANTENE
4. Tambahkan sebuah kolom PRODUKSI bertipe CHAR
5. Ubah tipe data kolom PRODUKSI menjadi INT
6. Buat satu buah database : COBA\_DROP
7. Buat satu buah tabel : COBA\_DROP ( coba int)
8. Hapus tabel COBA\_DROP
9. Hapus database COBA\_DROP

**TUGAS**

Gunakan data pada tugas sebelumnya :

1. Lakukan UPDATE isi sebuah kolom pada sebuah record.
2. Lakukan UPDATE isi 2 buah kolom pada sebuah record.
3. Hapus salah satu record
4. Tambahkan satu buah kolom baru pada tabel yang sudah ada
5. Isikan data pada kolom yang dibuat pada no.4
6. Buat satu buah database dan satu buah tabel di dalamnya
7. Lakukan penghapusan tabel dan database yang sudah dibuat pada no.6

**LATIHAN SOAL**

1. Buatlah database dengan nama **belajar**
2. Di Database belajar buatlah tabel dengan nama **barang** dengan field kode\_barang, nama\_barang, harga\_barang dan total\_barang silahkan tentukan sendiri tipe data dan panjangnya
3. Masukan data dibawah ini ke tabel barang tersebut

|  |  |  |  |
| --- | --- | --- | --- |
| **kode\_barang** | **nama\_barang** | **harga\_barang** | **total\_barang** |
| **1234** | **Laptop ASUS X453MA** | **4000000** | **100** |
| **1235** | **Macbook Air MD2015** | **14000000** | **20** |
| **1236** | **Printer Epson** | **1000000** | **40** |
| **1237** | **Flashdisk Kingston 8GB** | **80000** | **80** |
| **1238** | **Intel Core i5** | **2200000** | **10** |

1. Tampilkan barang dengan kode\_barang 1238
2. Ubah nama barang tersebut dengan nama **Intel Core i5 4594**
3. Ubah nama barang dengan nama **Flashdisk Kingston 8GB** menjadi **Kingston 8GB**
4. Tampilkan barang yang memiliki harga diatas 10 Juta
5. Hapus barang yang bernama **Printer Epson**