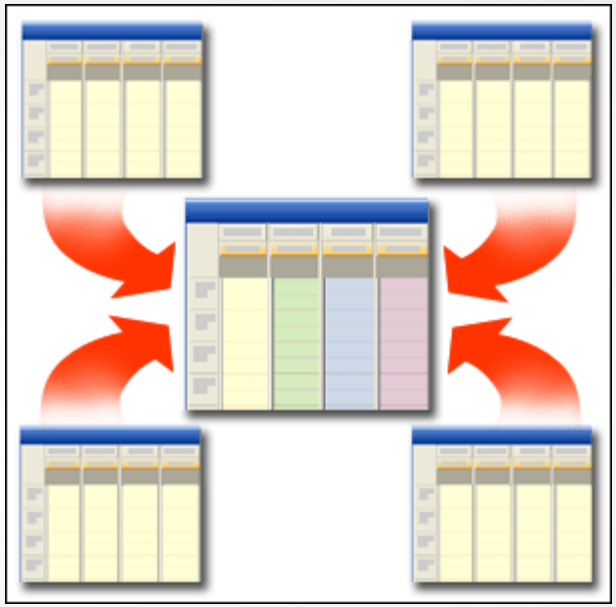
Praktikum Sistem Basis Data MYSQL

Gentisya Tri Mardiani, S.Kom., M.Kom



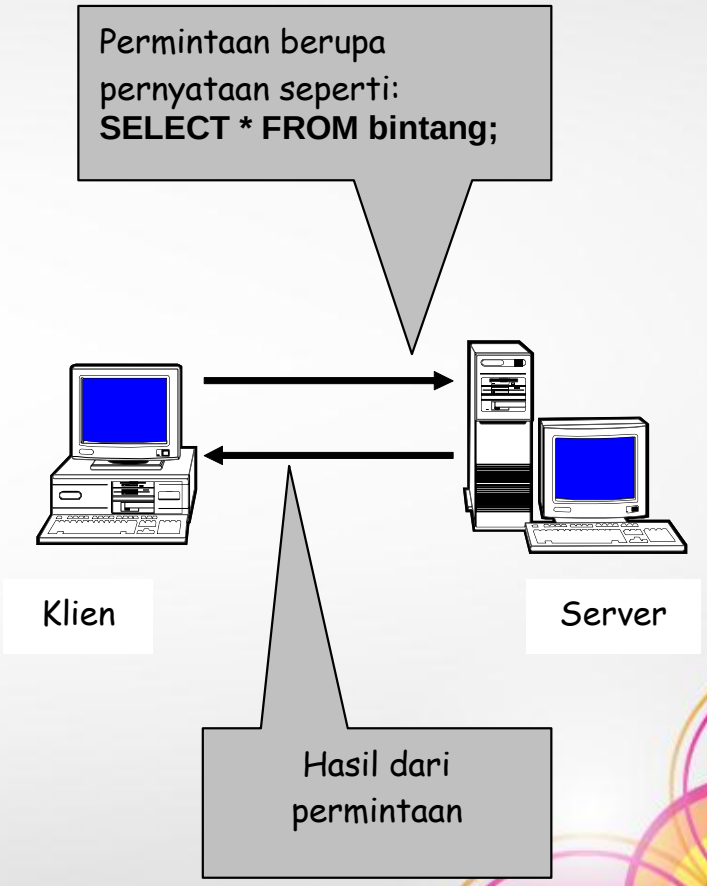
MySQL

- MySQL merupakan Database Server yang bersifat :
  - *Open Source*
  - Multiplatform
  - Berbasis *database* relasional
- Bisa dipakai untuk database pribadi atau pada level korporat berskala kecil hingga besar
- Selain bersifat free, ada juga yang bersifat komersial



Overview MySQL

- Menggunakan SQL untuk mendukung pengaksesan data (*query*)



Dasar SQL

- SQL = Structured Query Language
- Digunakan untuk mengakses basis data relasional
- Bersifat standar; bisa dipakai untuk basis data relasional lainnya
- Perintah SQL dapat dibagi menjadi DDL dan DML



Perintah DDL

- DDL = *Definition Data Language*
- Digunakan untuk kepentingan penciptaan database, tabel, hingga penghapusan database atau tabel
- Contoh:
  - CREATE DATABASE
  - CREATE TABLE
  - DROP TABLE
  - ALTER TABLE



Perintah DML

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



Persiapan Pemakaian MySQL

- Lakukan instalasi MySQL terlebih dulu
- MySQL memiliki sejumlah *tool*; salah satu di antaranya adalah:
  - program **mysql**, yang dipakai untuk mengakses *database* dari sisi klien
  - Program **mysqladmin**, untuk mengelola MySQL dari sisi administrator sistem

Masuk ke Direktori Program



Mengatur Password root

- **root** adalah pemakai dengan wewenang paling tinggi dan digunakan untuk admin sistem
- Perintah untuk mengubah password root:

  mysqladmin –uroot password rahasia

Masuk ke Program Klien mysql

- Berikan perintah:

  mysql –uroot –p

- Ketikkan password dan tekan Enter

```
C:\Program Files\MySQL\MySQL Server 5.0\bin>mysql -uroot -
prahasia
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 8 to server version: 5.0.15-nt

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

mysql>
```

Utilitas MySQL

- Untuk menampilkan perintah apa saja yang terdapat dalam mysql *client utility*, Anda dapat mengetikkan *help* di mysql *prompt*-nya.
- Hasilnya:

```
mysql> help
For information about MySQL products and services, visit:
   http://www.mysql.com/
For developer information, including the MySQL Reference Manual, visit:
   http://dev.mysql.com/
To buy MySQL Network Support, training, or other products, visit:
   https://shop.mysql.com/
List of all MySQL commands:
Note that all text commands must be first on line and end with ';'
?         (\?) Synonym for `help'.
clear     (\c) Clear command.
connect   (\r) Reconnect to the server. Optional arguments are db and host.
delimiter (\d) Set statement delimiter. NOTE: Takes the rest of the line as new delimiter.
ego       (\G) Send command to mysql server, display result vertically.
exit      (\q) Exit mysql. Same as quit.
go        (\g) Send command to mysql server.
help      (\h) Display this help.
notee     (\t) Don't write into outfile.
print     (\p) Print current command.
prompt    (\R) Change your mysql prompt.
quit      (\q) Quit mysql.
rehash    (\#) Rebuild completion hash.
source    (\.) Execute an SQL script file. Takes a file name as an argument.
status    (\s) Get status information from the server.
tee       (\T) Set outfile [to_outfile]. Append everything into given outfile.
use       (\u) Use another database. Takes database name as argument.
charset   (\C) Switch to another charset. Might be needed for processing binlog with multi-byte charsets.
warnings  (\W) Show warnings after every statement.
nowarning (\w) Don't show warnings after every statement.
For server side help, type 'help contents'
```

Menyimpan seluruh aktivitas

- Buat folder NIM Anda di direktori D:\

  tee D:\Namafolder\latihan1.txt

– Mendisfungsikan pencatatan ke file yang sedang dilakukan.

  notee

Pembuatan Database dan Tabel

Database: **Pegawai**

Tabel: **Pegawai**

Field:

-NIP

-Nama

-Tgl_Lahir

-Kelamin

-Alamat

-Kota

-Tgl_masuk

-Kode_jabatan

-Kode_area

Tabel: **Jabatan**

Field:

-Kode_Jabatan

-Nama_jabatan

-Gaji_pokok

Tabel: **Area**

Field:

-Kode_Area

-Nama_Area

-Alamat_Area

Membuat Database Pegawai

- Berikan perintah pada prompt mysql:

  SHOW DATABASES;

  CREATE DATABASE Pegawai;

  Contoh: CREATE DATABASE Pegawaisbd10;

```
mysql> CREATE DATABASE Pegawai;
Query OK, 1 row affected (0.09 sec)

mysql>
```

Memilih Database

- Berikan perintah:

  USE Pegawaisbd10;

Tipe Data

KARAKTER

- **CHAR**: Teks dengan maksimal 255 karakter
- **VARCHAR**: Teks maksimal 255 karakter dan bersifat variabel
- **TEXT**: Teks dengan panjang maksimal 65535



Tipe Data

BILANGAN

- **TINYINT**: Bilangan 1 byte
- **SMALLINT**: Bilangan 2 byte
- **INT** atau **INTEGER** Bilangan 4 byte
- **BIGINT**: Bilangan 8 byte
- **FLOAT**: Bilangan pecahan (4 byte)
- **DOUBLE** atau **REAL**: Bilangan pecahan (8 byte)
- **DECIMAL** atau **NUMERIC**: Bilangan pecahan



Tipe Data

LAIN-LAIN

- **DATE**: Tanggal (YYYY/MM/DD)
- **DATETIME**: Waktu (tanggal dan jam)

  (YYYY/MM/DD HH:MM:SS)
- **TIME :** Jam (HH:MM:SS)
- **ENUM(‘nilai1’, ‘nilai2’, …)**: Nilai enumerasi
- **BOOLEAN**: tipe benar atau salah



Kata Tambahan

- NOT NULL : Tidak boleh kosong
- PRIMARY KEY: Kunci primer
- AUTO_INCREMENT: Nilai naik secara otomatis tanpa diisi

Membuat Tabel Area Kerja

- Berikan perintah:

```
CREATE TABLE Area(
Kode_Area CHAR(2) NOT NULL PRIMARY KEY,
Nama_Area VARCHAR(35) NOT NULL,
Alamat_Area VARCHAR(35));
```



Melihat Tabel dalam database

- Gunakan perintah:

  SHOW TABLES;



Melihat Struktur Tabel

- Gunakan perintah:

  DESC nama_tabel
- Contoh:
- DESC Area;



Penambahan Data

- Penambahan data dilakukan dengan menggunakan pernyataan INSERT
- Bentuk dasar :

```
INSERT INTO nama_tabel (nama_field, nama_field,
    …) VALUES (nilai, nilai, …);
```

- Contoh:

```
INSERT INTO Area (Kode_Area, Nama_Area,
    Alamat_Area)
VALUES (‘G1', ‘Gedung 1 Pusat', 'Jl. Dipatiukur 35');
```

Melihat Isi Tabel

- Perintah SELECT

- Bentuk Umum:

  SELECT * FROM Nama_Tabel

- Contoh:

  SELECT * FROM Area;



Penambahan Data (Lanjutan…)

- Tambahkan data berikut:

Kode_area	Nama_area	Alamat_area
G2	Gedung 2	Jl. Ir.H.Juanda 100
G3	Gedung 3	Jl. Ir.H.Juanda 21
G4	Gedung 4	Jl. Siliwangi 50

Latihan Membuat Tabel Jabatan

- Buatlah tabel **Jabatan** dengan struktur seperti berikut:
  - **Kode_Jabatan**, CHAR, 2 karakter, primary key
  - **Nama_Jabatan**, VARCHAR, 20 karakter, harus diisi
  - **Gaji_Pokok**, bilangan
- Isikan data sebagai berikut:
  - 01 untuk EDP, gaji_pokok 1500000
  - 02 untuk Pemasaran , gaji_pokok 1200000
  - 03 untuk Produksi , gaji_pokok 2000000
  - 04 untuk SDM , gaji_pokok 2500000
  - 05 untuk Akunting , gaji_pokok 1200000

Latihan Membuat Tabel Pegawai

- Buatlah tabel **Pegawai** dengan struktur seperti berikut:

```
CREATE TABLE Pegawai(
NIP CHAR(5) NOT NULL PRIMARY KEY,
Nama VARCHAR(35) NOT NULL,
Tgl_lahir DATE,
Jenis_kelamin ENUM('P','W'),
Alamat VARCHAR(35),
Kota VARCHAR(15),
Tgl_masuk DATE,
Kode_Jabatan CHAR(2),
Kode_Area CHAR(2));
```



Membuat FOREIGN KEY

- **FOREIGN KEY**

- ALTER TABLE pegawai ADD FOREIGN KEY (Kode_Jabatan) REFERENCES jabatan (Kode_Jabatan);

- ALTER TABLE pegawai ADD FOREIGN KEY (Kode_Area) REFERENCES area(Kode_Area);



- Isikan data sebagai berikut:

NIP	Nama	Tgl_lahir	Jenis_ kelamin	Alamat	Kota	Tgl_masuk	Kode_ja batan	Kode _area
12346	Udin	22/01/1978	P	Jl.Masjid 47	Sleman	02/02/1999	02	G1
12347	Arum Dian	14/03/1980	W	Jl.Sawo 108	Yogya	02/02/1999	01	G2
12348	Sueb	04/07/1971	P	Jl.Astina 4A	Yogya	02/02/1999	02	G1
12349	Bagus	13/05/1969	P	Jl.Karangwaru 3	Magelang	02/02/1999	04	G1
12350	Intan	01/02/1975	W	Jl.Karyacita 9	Bantul	02/02/1999	05	G3



Penambahan Data dengan Field Tertentu

- Contoh tanpa mengisi Alamat

```
INSERT INTO Area(Kode_Area, Nama_Area)
VALUES (‘G5’,‘Gedung 5’);
```

Pengaruh Kunci Primer

- Cobalah berikan pernyataan berikut:

```
INSERT INTO Area
   (Kode_Area,Nama_Area,Alamat_Area)
VALUES (‘G2’,’Gedung 2 baru’,’Jl.Ir.H.Juanda 150');
```

- Apa yang terjadi? Kenapa?



Pengaruh NOT NULL

- Cobalah berikan pernyataan berikut:

```
INSERT INTO Area(Kode_Area, Alamat_Area)
VALUES (‘G7', ‘Jl. Dipatiukur 160');
```

- Apa yang terjadi? Kenapa?

Pengaruh ENUM

- Cobalah perintah berikut:

```
INSERT INTO pegawai (NIP, Nama, Jenis_kelamin)
VALUES ('12361', 'Edi harahap', 'L');
```

- Apa yang terjadi? Kenapa?
- Cek dengan SELECT; apakah data tersimpan?

Mengganti Nama Field

- Perintah yang digunakan adalah ALTER TABLE
- Contoh:

  ALTER TABLE Pegawai

  CHANGE Jenis_kelamin Kelamin ENUM('P','W');

- Cek struktur tabel setelah Anda melakukan perintah di atas
- DESC Pegawai;



Mengganti Ukuran/Tipe Field

- Perintah yang digunakan adalah ALTER TABLE
- Contoh:

  ALTER TABLE Pegawai

  CHANGE Kota Kota VARCHAR(20);

- Cek struktur tabel setelah Anda melakukan perintah di atas
- DESC Pegawai;

Menambahkan DEFAULT

- DEFAULT pada struktur tabel digunakan untuk memberikan nilai bawaan /default pada suatu field jika nilai tersebut tidak dimasukkan
- Contoh:

  ALTER TABLE Pegawai CHANGE Kelamin Kelamin ENUM('P','W') DEFAULT 'P';

Menambahkan DEFAULT (Lanjutan…)

- Sekarang cobalah berikan perintah:

  INSERT INTO Pegawai(Nip, Nama)

  VALUES (‘12370’, ‘Fahmi Idris’);

- PERHATIKAN, apa isi field Kelamin untuk NIP 12370’?

Pengubahan Data

- Perintah yang digunakan adalah UPDATE
- Bentuk dasar:

  UPDATE nama_tabel SET nama_field = nilai … WHERE nama_field = nilai

- Contoh mengubah **Udin** menjadi **Udinsah**:

  UPDATE Pegawai SET Nama= 'Udinsah'
  WHERE NIP = '12346';

- Ujilah dengan SELECT untuk melihat hasil perubahan



Penghapusan Data

- Perintah DELETE
- Bentuk dasar:
  DELETE FROM nama_tabel WHERE nama_field = nilai

- Contoh:
  DELETE FROM Pegawai
  WHERE Nip = '12355';

- Ujilah dengan SELECT untuk melihat hasil perintah di atas

Penghapusan Tabel

- Gunakan DROP TABLE
- Untuk mempraktekkan, buatlah sebuah table bernama RIWAYAT, dengan isi berupa sebuah field (misalnya Nip bertipe CHAR(5))
- Kemudian, lihatlah daftar tabel dengan memberikan perintah:

  SHOW TABLES;



Penghapusan Tabel (Lanjutan…)

- Berikan perintah:

  DROP TABLE Riwayat;

- Kemudian, lihatlah daftar tabel dengan memberikan perintah:

  SHOW TABLES;

  Apa sudah terhapus?



Latihan Penggunaan Tipe Memo

- Buatlah tabel bernama **Riwayat**
- Isi Field:
  - NIP CHAR 5 karakter, PRIMARY KEY
  - Keterangan (bertipe Memo)
- Isikan data untuk NIP yang tersedia pada tabel Pegawai (12346 sampai dengan 12350)
- Isikan Keterangan dengan daftar riwayat pendidikan, misalnya sbb:

SD Muhammadiyah I, Yogya
SMP Negeri I, Yogya
SMA Negeri III, Semarang



Latihan

- Masukkan data berikut ke Tabel Pegawai:
  - NIP: 12390
  - Nama: Asti Damayanti
  - Tanggal Lahir: 2 Februari 1983
  - Jenis kelamin: Wanita
  - Tanggal masuk: 2 februari 2002
  - Kode area: G1
  - Kode jabatan: 04

- Ubahlah Tanggal lahir Asti Damayanti menjadi 23 Februari 1973
- Cek hasilnya
- Hapuslah record yang berisi data Asti Damayanti dengan menyebutkan namanya (bukan NIP)