

LAPORAN AKHIR
PENGARSIPAN DATA PERSONIL DITLANTAS POLDA
SULUT MENGGUNAKAN MYSQL
DIREKTORAT LALU LINTAS (DITLANTAS) POLDA SULUT



OLEH :

WELMUS NALLY TAMERA

12 022 014

DOSEN PEMBIMBING

EDWIN STEPHANUS ALLEN LUMUNON,ST,MIT

NIP. 19740121 199802 1 001

KEMENTERIAN RISET TEKNOLOGI DAN PENDIDIKAN TINGGI
POLITEKNIK NEGERI MANADO
JURUSAN TEKNIK ELEKTRO
PROGRAM STUDI D III KOMPUTER
2015

HALAMAN PENGESAHAN
PENGARSIPAN DATA DITLANTAS POLDA SULUT MENGGUNAKAN
MYSQL
DIREKTORAT LALU LINTAS (DITLANTAS) POLDA SULUT

Oleh

Welmus Nally Tamera

12 022 014

Laporan Akhir ini telah diterima dan disahkan sebagai persyaratan

Menyelesaikan pendidikan Diploma III Teknik Elektro

Program Studi Teknik Komputer

Politeknik Negeri Manado

Manado, 8 Agustus 2015

Menyetujui :

Ketua Panitia Laporan Akhir,

Dosen Pembimbing,

Fanny J, Doringin. ST. MT
NIP.19670430 199203 1 003

Edwin Stephanus Allen Lumunon,ST,MIT
NIP. 19740121 199802 1 001

Mengetahui
Ketua Jurusan Teknik Elektro

Ir. Luther Mappadang, MT.
NIP. 19610601199003 1 001

KATA PENGANTAR

Puji dan syukur Penyusun panjatkan kepada Tuhan Yesus Kristus sebagai sumber kekuatan, hikmat, pengetahuan dan kemampuan, sehingga Penyusun dapat menyelesaikan Studi kasus dengan judul “ Pengarsipan Data Personil Ditlantas Polda Sulut Menggunakan MySQL”. Dengan harapan dari penulis semoga pembuatan studi kasus ini dapat berguna dan juga bisa membantu pekerjaan Ditlantas Polda Sulut kedepannya.

Dalam penyusunan studi kasus ini penulis mengalami banyak hambatan dan kesulitan namun berkat pertolongan Tuhan Yang Maha Esa serta dorongan dan bantuan semua pihak, baik langsung maupun tidak langsung maka studi kasus ini dapat diselesaikan dengan baik .

Penulis tak lupa menyampaikan terima kasih kepada:

1. Orang Tua
2. Kasubbag Renmin Kompol Ismael .T. Jacobus
3. Ibu Selfi Kaur tu
4. Iptu Repi
5. Ipda Romi
6. Briptu kadek
7. Dana Wiwin Taluke
8. Dosen
9. Teman-teman yang memberikan motifasi

Penyusun menyadari bahwa dalam pembuatan srtudi kasus ini masih banyak kesalahan dan kekurangan yang ada sehingga penyusun sangat menerima kritikan,masukan dan saran dari seluruh pihak demi perbaikan penyusunan studi kasus ini. Kiranya pembuatan studi kasus ini dapat memberikan manfaat bagi para pembaca. Atas perhatian dan kerjasama yang baik saya ucapkan terima kasih Tuhan Yesus Memberkati.

Manado,Juli 2015

Welmus Nally Tamera

DAFTAR ISI

Judul.....	i
Lembar Pengesahan.....	ii
Kata Pengantar.....	iii
Daftar Isi.....	v
BAB I. PENDAHULUAN.....	1
1. 1 Latar Belakang.....	1
1. 2 Tujuan Penulisan.....	2
1. 3 Ruang Lingkup Studi Kasus.....	2
1. 4 Perumusan Masalah Studi Kasus.....	3
1. 5 Kegunaan Studi Kasus.....	3
BAB II. LANDASAN TEORI.....	4
2. 1 Pengertian Database.....	4
2. 2 Beberapa Defenisi Tentang Database.....	4
2. 3 Defenisi Dasar Struktur Database.....	5
2. 4 Sifat-sifat Database.....	6
2. 5 Tipe Database.....	7
2. 6 Model Database.....	9
2. 7 Tujuan Database.....	9
2. 8 Ciri-ciri Database.....	10
2. 9 Keuntungan Database.....	10
2. 10 Manfaat Database.....	10
BAB III. PEMBAHASAN STUDI KASUS.....	12
3. 1 Program Yang Dugunakan Untuk Mebuat System Data Base Yaitu MySQL.....	12
3. 2 Mengapa Memilih MySQL.....	13
3. 3 Kelebihan Dan Kelemahan MySQL.....	14
3. 4 Proses Instalasi MySQL.....	16
3. 5 Pembuatan Database.....	24
BAB IV. KESIMPULAN DAN SARAN	35

4.1 Bagian Akhir.....	35
-----------------------	----

BAB I

PENDAHULUAN

1.1. Latar Belakang

Basis data adalah kumpulan informasi yang disimpan di dalam komputer secara sistematis sehingga dapat diperiksa menggunakan suatu program komputer untuk memperoleh suatu informasi dari database tersebut. Perangkat lunak yang digunakan untuk mengolah dan mengambil query basis data disebut sistem manajemen basis data. Pemrosesan basis data sebagai perangkat andalan sangat diperlukan oleh berbagai institusi dan perusahaan. Dalam pengembangan sistem informasi diperlukan basis data sebagai media penyimpanan data. Kehadiran basis data dapat meningkatkan daya saing perusahaan tersebut. Basis data dapat mempercepat upaya pelayanan kepada pelanggan, menghasilkan informasi dengan cepat dan tepat sehingga membantu pengambilan keputusan untuk segera memutuskan suatu masalah berdasarkan informasi yang ada. Banyak aplikasi yang dibuat dengan berlandaskan pada basis data antara lain semua transaksi perbankan, aplikasi pemesanan, penjadwalan penerbangan, proses registrasi dan pencatatan data mahasiswa pada perguruan tinggi, aplikasi pemrosesan penjualan, pembelian dan pencatatan data barang pada perusahaan dagang, pencatatan data pegawai beserta aktivitasnya termasuk operasi penggajian pada suatu perusahaan, dan sebagainya. Beberapa informasi pada perusahaan retail seperti jumlah penjualan, mencari jumlah stok yang tersedia, barang apa yang paling laku dijual pada bulan ini, dan berapa laba bersih perusahaan dapat diketahui dengan mudah menggunakan basis data.

Pentingnya studi kasus bertujuan supaya mahasiswa mampu memecahkan masalah yang ada dalam suatu instansi atau perusahaan. Penulis memilih DITLANTAS POLDA SULUT sebagai tempat Praktek Kerja Lapangan.

Ditlantas Polda Sulut selalu mengutamakan keselamatan berlalu lintas di jalan sehingga dibuat beberapa operasi dalam satu tahun ada empat operasi yang dilakukan oleh setiap polres-polres yang ada di Sulawesi utara.

Masalah yang penulis dapat belum ada sistem database Ditlantas Polda Sulut yang baik

Untuk itu penulis melaksanakan studi kasus ini dengan maksud untuk merancang sebuah sistem database tentang Pengarsipan Personil Ditlantas Polda Sulut.

1.2. Tujuan penulisan

Studi kasus ini dilaksanakan dalam usaha untuk menguasai cara menggunakan program database dan membuat sebuah sistem database tentang Ditlantas Polda Sulut untuk melatih dan menguji kemampuan berfikir kritis, kreatif dan analitis untuk memperkaya ilmu pengetahuan teoritis yang diperoleh mahasiswa di bangku kuliah dengan pengalaman-pengalamannya selama melakukan praktek kerja lapangan dilapangan, sehingga mahasiswa mampu :

1. Mendeskripsikan suatu permasalahan
2. Mengkaitkan permasalahan tersebut dalam bidang ilmu manajemen Komputer
3. Mendeteksi permasalahan yang sedang atau akan terjadi
4. Mengambil kesimpulan dan saran dari analisis permasalahan tersebut di atas.

1.3. Ruang Lingkup Studi Kasus

Penulis melihat di Ditlantas Polda Sulut belum ada sistem database yang baik tentang pengarsipan data personil . Untuk itu penulis berencana membuat sebuah sistem database untuk pengarsipan data personil Ditlantas Polda Sulut menggunakan program MySQL dengan harapan dapat membantu Ditlantas Polda Sulut dalam menjalankan pekerjaannya.

1.4. Perumusan Masalah Studi Kasus

Berdasarkan latar belakang yang telah diuraikan di atas, maka dapat diambil rumusan masalah dalam studi kasus ini adalah:

- 1). Apa jenis program yang akan dipakai untuk membuat sistem data base Ditlantas Polda Sulut dan tingkatan masalah Ditlantas?

- 2). Apa faktor penyebab masalah yang didapat dalam membuat system data Base Polda Sulut ?
- 3). Bagaimana rencana dalam mengatasi masalah dalam membuat system data base ?

1.5. Kegunaan Studi Kasus

Studi kasus sendiri merupakan metode pengumpulan data yang bersifat menyeluruh dan terpadu. Menyeluruh berarti data yang dikumpulkan meliputi seluruh aspek pribadi individu. Terpadu artinya menggunakan berbagai pendekatan dalam mengumpulkan data. Dengan demikian dari data yang terkumpul akan diperoleh pemahaman individu yang dimaksud. Individu yang dikenakan dalam studi kasus menunjukkan gejala mengalami kesulitan atau masalah yang serius, sehingga membutuhkan bantuan yang secepatnya. Penulis menyusun studi kasus ini supaya dapat digunakan dan berguna di Ditlantas Polda Sulut.

BAB II

LANDASAN TEORI

2.1. Pengertian Database

Istilah “database” berawal dari ilmu komputer. Meskipun kemudian artinya semakin luas, memasukkan hal-hal di luar bidang elektronika, artikel ini mengenai database komputer. Catatan yang mirip dengan database sebenarnya sudah ada sebelum revolusi industri yaitu dalam bentuk buku besar, kuitansi dan kumpulan data yang berhubungan dengan bisnis.

Database (basis data) adalah kumpulan file-file yang mempunyai kaitan antara satu file dengan file yang lain sehingga membentuk data untuk menginformasikan satu perusahaan dan instansi. Bila terdapat file yang tidak dapat dipadukan atau dihubungkan dengan file yang lainnya, berarti file tersebut bukanlah kelompok dari satu database, melainkan membentuk satu database sendiri. Database juga merupakan landasan bagi pembuatan dan pengembangan program aplikasi. Oleh sebab itu, database harus dibuat sedemikian rupa sehingga pembuatan program lebih mudah dan cepat.

Database merupakan salah satu komponen yang penting dalam sistem informasi, karena merupakan basis dalam menyediakan informasi bagi para pemakai. Database terdiri dari data yang akan digunakan atau diperuntukkan terhadap banyak *user*, dari masing-masing *user* akan menggunakan data tersebut sesuai dengan tugas dan fungsinya.

2.2. Beberapa Definisi tentang Database

1. Menurut Gordon C. Everest :

Database adalah koleksi atau kumpulan data yang mekanis, terbagi / shared, terdefinisi secara formal dan dikontrol terpusat pada organisasi.

2. Menurut C.J. Date :

Database adalah koleksi “data operasional” yang tersimpan dan dipakai oleh sistem aplikasi dari suatu organisasi.

- Data input adalah data yang masuk dari luar system
- Data output adalah data yang dihasilkan sistem
- Data operasional adalah data yang tersimpan pada sistem

3. Menurut Toni Fabbri :

Database adalah sebuah sistem file-file yang terintegrasi yang mempunyai minimal primary key untuk pengulangan data.

4. Menurut S. Attre :

Database adalah koleksi data-data yang saling berhubungan mengenai suatu organisasi / enterprise dengan macam-macam pemakaiannya.

2.3. Definisi Dasar Struktur Database

- **Data:** Sekumpulan fakta mengenai objek tertentu, orang dan lain-lain yang dinyatakan dengan angka, huruf, gambar, film, suara dan sebagainya yang relevan dan belum mempunyai arti.
- **Informasi:** Hasil pengolahan data yang konkrit dan sudah mempunyai arti untuk mencapai suatu tujuan tertentu.
- **Tabel:** Merupakan hal yang paling mendasar dalam hal penyimpanan data yang terdiri dari field dan record.
- **Field (kolom):** Merupakan elemen dari tabel yang berisikan informasi tertentu yang spesifik tentang subjudul tabel pada sebuah item data.

Syarat-syarat pembentukan Field Name pada tabel:

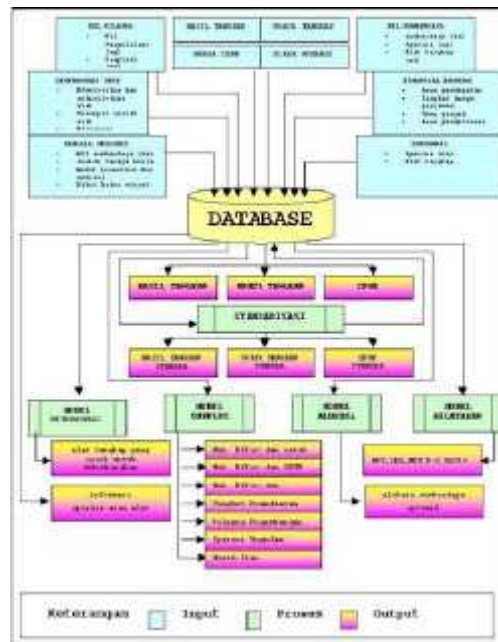
- a) Harus Unik atau Spesifik
- b) Boleh disingkat
- c) Pemisah sebagai pengganti spasi dalam pembentuk field adalah tanda lambang "_"

Contoh: Kode Barang menjadi KdBarang, KodeBrg, Kd_Brg, Kd_Barang.

- **Record (baris):** Sekumpulan data yang saling berkaitan tentang sebuah subjek tertentu, misalnya data seorang siswa akan disimpan dalam record yang terdiri dari beberapa kolom / field.

2.4. Sifat-sifat Database

- Internal: Kesatuan (integritas) dari file-file yang terlibat.
- Terbagi/share: Elemen-elemen database dapat dibagikan pada para user baik secara sendiri-sendiri maupun secara serentak dan pada waktu yang sama (*Concurrent sharing*).



Perangkat Untuk Membuat Database

Database dapat dibuat dan diolah dengan menggunakan suatu program komputer, yaitu yang biasa kita sebut dengan software (perangkat lunak). Software yang digunakan untuk mengelola dan memanggil kueri (*query*) database disebut Database Management System (DBMS) atau jika diterjemahkan kedalam bahasa Indonesia berarti “Sistem Manajemen Basis Data”.

2.5. Tipe Database

Terdapat 12 tipe database, antara lain:

1. **Operational database:** Database ini menyimpan data rinci yang diperlukan untuk mendukung operasi dari seluruh organisasi. Mereka juga disebut *subject- area databases* (SADB), transaksi database, dan produksi database. Contoh: database pelanggan, database pribadi, database inventaris, akuntansi database.
2. **Analytical database:** Database ini menyimpan data dan informasi yang diambil dari operasional yang dipilih dan eksternal database. Mereka terdiri dari data dan informasi yang dirangkum paling dibutuhkan oleh sebuah organisasi manajemen dan End-user lainnya. Beberapa orang menyebut analitis multidimensi database sebagai database, manajemen database, atau informasi database.
3. **Data warehouse:** Sebuah data warehouse menyimpan data dari saat ini dan tahun- tahun sebelumnya - data yang diambil dari berbagai database operasional dari sebuah organisasi.
4. **Distributed database:** Ini adalah database-kelompok kerja lokal dan departemen di kantor regional, kantor cabang, pabrik-pabrik dan lokasi kerja lainnya. Database ini dapat mencakup kedua segmen yaitu operasional dan user database, serta data yang dihasilkan dan digunakan hanya pada pengguna situs sendiri.
5. **End-user database:** Database ini terdiri dari berbagai file data yang dikembangkan oleh end-user di workstation mereka. Contoh dari ini adalah koleksi dokumen dalam spreadsheet, word processing dan bahkan download file.
6. **External database:** Database ini menyediakan akses ke eksternal, data milik pribadi online - tersedia untuk biaya kepada pengguna akhir dan organisasi dari layanan komersial. Akses ke kekayaan informasi dari database eksternal yang tersedia untuk biaya dari layanan online komersial dan dengan atau tanpa biaya dari banyak sumber di Internet.

7. **Hypermedia databases on the web:** Ini adalah kumpulan dari halaman-halaman multimedia yang saling berhubungan di sebuah situs web. Mereka terdiri dari home page dan halaman hyperlink lain dari multimedia atau campuran media seperti teks, grafik, gambar foto, klip video, audio dll.
8. **Navigational database:** Dalam navigasi database, queries menemukan benda terutama dengan mengikuti referensi dari objek lain.
9. **In-memory databases:** Database di memori terutama bergantung pada memori utama untuk penyimpanan data komputer. Ini berbeda dengan sistem manajemen database yang menggunakan disk berbasis mekanisme penyimpanan. Database memori utama lebih cepat daripada dioptimalkan disk database sejak Optimasi algoritma internal menjadi lebih sederhana dan lebih sedikit CPU mengeksekusi instruksi.
10. **Document-oriented databases:** Merupakan program komputer yang dirancang untuk aplikasi berorientasi dokumen. Sistem ini bisa diimplementasikan sebagai lapisan di atas sebuah database relasional atau objek database. Sebagai lawan dari database relasional, dokumen berbasis database tidak menyimpan data dalam tabel dengan ukuran seragam kolom untuk setiap record. Sebaliknya, mereka menyimpan setiap catatan sebagai dokumen yang memiliki karakteristik tertentu. Sejumlah bidang panjang apapun dapat ditambahkan ke dokumen. Bidang yang dapat juga berisi beberapa bagian data.
11. **Real-time databases Real-time:** Database adalah sistem pengolahan dirancang untuk menangani beban kerja negara yang dapat berubah terus-menerus. Ini berbeda dari database tradisional yang mengandung data yang terus-menerus, sebagian besar tidak terpengaruh oleh waktu. Sebagai contoh, pasar saham berubah dengan cepat dan dinamis. Real-time processing berarti bahwa transaksi diproses cukup cepat bagi hasil untuk kembali dan

bertindak segera. Real-time database yang berguna untuk akuntansi, perbankan, hukum, catatan medis, multi-media, kontrol proses, sistem reservasi, dan analisis data ilmiah.

12. **Relational Database:** Database yang paling umum digunakan saat ini. Menggunakan meja untuk informasi struktur sehingga mudah untuk mencari.

2.6. Model Database

Database mempunyai dua varian model, yaitu model Post-relational database dan model Object database.

1. Post-relational database models

Sebuah produk yang menawarkan model data yang lebih umum dari model relasional dan dikenal sebagai post-relational. Model data dalam produk tersebut mencakup hubungan namun tidak dibatasi oleh Prinsip Informasi yang mana mewakili semua informasi dengan nilai-nilai data dalam kaitannya dengan hal itu.

2. Object database models

Dalam beberapa tahun terakhir,[update], paradigma yang berorientasi pada obyek telah diterapkan dalam bidang-bidang seperti teknik dan spasial database, telekomunikasi dan ilmu ilmiah lainnya. Para konglomerasi pemrograman berorientasi objek dan teknologi database mengarah pada model pemrograman baru yang dikenal sebagai Object database.

2.7. Tujuan Database

Setiap manajemen dalam merancang dan menyusun database harus mempunyai tujuan, yaitu:

1. Membuat agar *user* mudah mendapatkan data.
2. Menyediakan tempat penyimpanan data yang relevan.
3. Menghapus data yang berlebihan.
4. Melindungi data dari kerusakan fisik.

5. Memungkinkan perkembangan lebih lanjut di dalam sistem database.

2.8. Ciri-ciri Database

Adapun ciri-ciri database adalah:

1. Sistem yang dapat menyimpan data ke dalam *floppy disk* atau *harddisk*.
2. Sistem yang menganut pengolahan data untuk ditambah, diubah, atau dihapus dengan mudah dan terkontrol.
3. Data terpisah dari program.

2.9. Keuntungan Database

Adapun keuntungan database terhadap sistem pemrosesan adalah:

1. Kemubajiran data berkurang.
2. Integritas data dapat selalu terjaga.
3. Berbagai data dapat selalu dilakukan oleh setiap *user*.
4. Penggunaan data lebih mudah.
5. Konsistensi data dapat selalu terjaga.

2.10. Manfaat Database

Adapun manfaat database adalah:

1. Sebagai komponen utama atau penting dalam sistem informasi, karena merupakan dasar dalam menyediakan informasi.
2. Menentukan kualitas informasi yaitu cepat, akurat, dan relevan, sehingga informasi yang disajikan tidak basi. Informasi dapat dikatakan bernilai bila manfaatnya lebih efektif dibandingkan dengan biaya mendapatkannya.
3. Mengatasi kerangkapan data (*redundancy data*).
4. Menghindari terjadinya inkonsistensi data.
5. Mengatasi kesulitan dalam mengakses data.
6. Menyusun format yang standar dari sebuah data.

7. Penggunaan oleh banyak pemakai (*multiple user*). Sebuah database bisa dimanfaatkan sekaligus secara bersama oleh banyak pengguna (*multiuser*).
8. Melakukan perlindungan dan pengamanan data. Setiap data hanya bisa diakses atau dimanipulasi oleh pihak yang diberi otoritas dengan memberikan login dan password terhadap masing-masing data.
9. Agar pemakai mampu menyusun suatu pandangan (*view*) abstraksi dari data. Hal ini bertujuan menyederhanakan interaksi antara pengguna dengan sistemnya dan database dapat mempresentasikan pandangan yang berbeda kepada para pengguna, programmer dan administratornya.

BAB III

PEMBAHASAN STUDI KASUS

Adapun hal-hal yang dibahas pada bab ini meliputi hasil pengamatan di Ditlantas Polda Sulut ada beberapa masalah yang ada di Ditlantas Polda Sulut yang penulis tonjolkan yaitu kurangnya system pengarsipan database yang baik . Karena itu penulis mengambil judul ini untuk membuat system pengarsipan database personil Ditlantas Polda Sulut.

3.1. Program Yang Digunakan Untuk Membuat System Database Yaitu (MySQL)



Pengertian Mysql

MySQL adalah sebuah perangkat lunak system manajemen basis data SQL (DBMS) yang multithread, dan multi-user. MySQL adalah implementasi dari system manajemen basisdata relasional (RDBMS). MySQL dibuat oleh TcX dan telah dipercaya mengelola system dengan 40 buah database berisi 10.000 tabel dan 500 di antaranya memiliki 7 juta baris.

kali diterapkan pada sebuah proyek riset pada laboratorium riset San Jose, IBM yang bernama system R. Kemudian SQL juga dikembangkan oleh Oracle, Informix dan Sybase.

3.2. Mengapa Memilih MySQL

Karena penulis lebih memahami dan mengerti system kerja pembuatan database dengan menggunakan Mysql dari pada menggunakan programlain.dan Mysql bisa di dawnload secara gratis di internet. Mysql juga memiliki banyak hal yang bisadi tawarkan antara lain:

Kelebihan

Banyak ahli berpendapat Mysql merupakan server tercepat.

- Kemudahan penggunaan

Mysql punya performa tinggi namun merupakan database yang simple sehingga mudah disetup dan dikonfigurasi.

- Harga

MySQL cenderung gratis dari penggunaan tertentu.

- Mendukung query language

MySQL mengerti bahasa SQL (Structured Query Language) yang merupakan pilihan system database modern.Juga dapat mengakses MySQL lewat protocol ODBC (Open Database Connectivity) buatan Microsoft.

- Kapabilitas

Banyak klien dapat mengakses server dalam suatu waktu.Dapat menggunakan banyak database secara simultan.

- Konektifitas dan sekuritas

Database MySQL dapat diakses dari semua tempat Internet dengan hak akses tertentu.

- Pertabilitas

MySQL dapat berjalan dalam banyak varian UNIX dang baik, sebaik seperti saat berjalan di system non-UNIX.

- Distribusi yang terbuka

MySQL mudah didapat dan memiliki source code yang bisa disebarluarkan sehingga bisa di kembangkan lebih lanjut.

3.3. Kelebihan Dan Kekurangan MySQL

Kelebihan :

- Merupakan DBMS yang gratis / open source berlisensi GPL (generic public license).
- Cocok untuk perusahaan dengan skala yang kecil.
- Tidak membutuhkan spesifikasi hardware yang tinggi untuk bisa menjalankan MWSQL ini bahkan dengan spesifikasi hardware yang minimal sekalipun.
- Bisa berjalan pada lebih dari satu platform system operasi, misalnya LINUX, Windows, MacOS, FreeBSD, Solaris dan masih banyak lagi.
- Cepat dalam menjalankan perintah *SQL / Structured Query Language* misalnya ketika akan menyeleksi suatu data atau memasukkan suatu data karena MYSQL merupakan turunan dari konsep *SQL.Multi user*, artinya database dapat digunakan oleh beberapa user dalam waktu bersamaan tanpa mengalami masalah atau konflik.
- MYSQL memiliki ragam tipe data yang sangat kaya, seperti signed / unsigned integer yang memiliki panjang data sebesar 1,2,3,4 dan 8 byte, FLOAT, DOUBLE, CHAR, VARCHAR, TEXT, BLOB, DATE, TIME, DATETIME, TIMESTAMP, YEAR, SET dan tipe ENUM.
- MYSQL memiliki beberapa lapisan keamanan, seperti *subnetmask*, *namahost*, dan izin akses user dengan system perijinan yang mendetail serta sandi/password terenkripsi.
- MYSQL dapat melakukan koneksi dengan computer client menggunakan Protokol TCP/IP, Unix Socket (UNIX), atau Named Pipes (windows NT).
- MYSQL memiliki antar muka / *interface* terhadap berbagai aplikasi dan bahasa pemrograman dengan menggunakan fungsi API (*Application Programming Interface*).
- *Command and function*, MYSQL memiliki fungsi dan operator secara penuh yang mendukung perintah *select* dan *where* dalam query.
- *Structure Table*, MYSQL memiliki struktur tabel yang lebih fleksibel dalam menangani *ALTER TABLE* dibandingkan DBMS lainnya.
- Dapat dikoneksikan pada bahasa C, C++, Java, Perl, PHP dan Python.

- Mendukung penuh terhadap kalimat SQL GROUP BY dan ORDER BY. Mendukung terhadap fungsi penuh (COUNT (),COUNT (DISTINCT), AVG (), STD (), SUM (), MAX () AND MIN ()).
- Mendukung terhadap LEFT OUTER JOIN dengan ANSI SQL dan sintak ODBC.
- Mendukung ODBC for Windows 95 (dengan source program). Semua fungsi ODBC 2.5 dan sebagainya. Sebagai contoh kita dapat menggunakan Access untuk connect ke MySQL server.
- Menggunakan GNU automake, autoconf, dan LIBTOOL untuk portabilitas.
- Kita dapat menggabungkan beberapa table dari database yang berbeda dalam query yang sama.
- Privilege (hak) dan password sangat fleksibel dan aman serta mengujinkan 'Host-Based' Verifikasi.

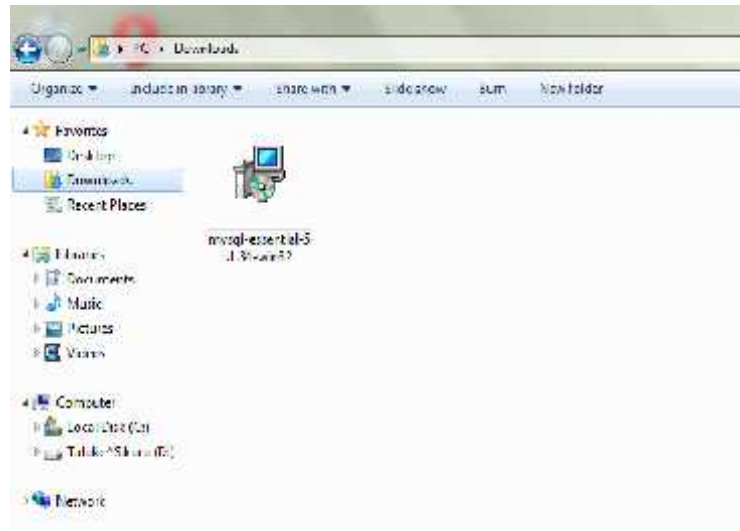
Kekurangan :

- Tidak cocok untuk menangani data dengan jumlah yang besar, baik untuk menyimpan data maupun untuk memproses data.
- Memiliki keterbatasan kemampuan kinerja pada server ketika data yang disimpan telah melebihi batas maksimal kemampuan daya tampung server karena tidak menerapkan konsep *Technology Cluster Server*.
- Untuk koneksi ke bahasa pemrograman visual seperti vb, delphi, dan foxpro, mysql kurang support, karena koneksi ini menyebabkan field yang dibaca harus sesuai dengan koneksi dari program visual tersebut, dan ini yang menyebabkan mysql jarang dipakai dalam program visual.

3.4. Proses Instalasi MySQL

- Langkah 1

Pilih dan sorotlah MySQL yang baru di download nama filenya adalah mysql-essential-5.1.34-win32.



Gambar 1.1 File MySQL yang telah di download

- Langkah 2

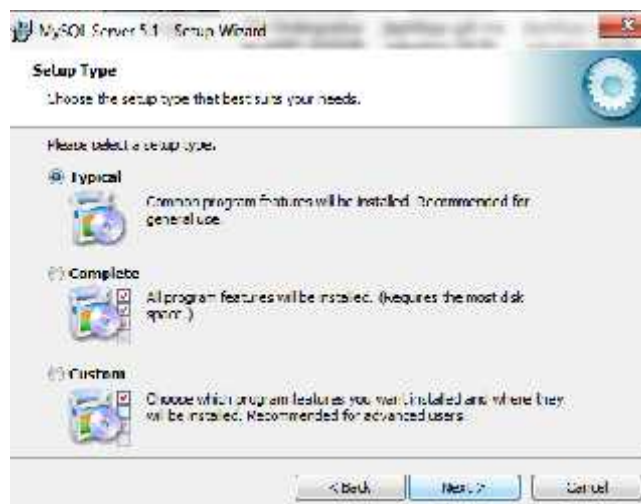
Double Click file tersebut, maka muncul tampilan selamat datang. Untuk melanjutkan proses instalasi klik pada tombol *Next*>



Gambar 1.2 MySQL server 5.1-setup wizard (1)

Langka 3

Pada tampilan ini Anda harus memilih jenis instalasi yang akan dilaksanakan. Yang paling mudah adalah jenis instalasi Typical. Maka saya menggunakan Typical saja... :) Untuk melanjutkan, klik pada tombol Next >



Gambar 1.3 MySQL Server 5.5-setup wizard

Langka 4

Tampilan berikutnya konfirmasi bahwa program MySQL akan diinstall ke dalam direktori C:\Program Files\MySQL\MySQL Server 5.1\ . Karena kitamenggunakan jenis instalasi Typical, maka direktori instalasi tidak

dapat kita ubah (kecuali menggunakan jenis instalasi Custom pada langkah no.3 di atas). Untuk sementara sayagunakan saja apa adanya. Dan klik tombol Install untuk melanjutkan.

Langka 5

Santai dulu sambil menunggu proses persiapan instalasi berlangsung... :)



Gambar 1.4 MySQL Server 5.1-Setup Wizard (3)

Langka 6

Tampilan berikutnya, informasi mengenai produk MySQL Enterprise(harus membayar sejumlah uang untuk menggunakan produk berlisensi ini). Bila berminat untuk menggunakan produk dengan lisensi ini, silakan kunjungi situsnya. Mari kita lanjutkan proses instalasi ini dengan menekan tombol Next >sebanyak dua kali.



Gambar 1.5 MySQL Enterprise (1)



Gambar 1.6 MySQL Enterprise (2)

Langka 7

Tunggu beberapa saat hingga proses instalasi selesai. Kemudian (sangat disarankan) untuk melanjutkan ke proses konfigurasi MySQL server (Configure the MySQL Server now). Dan bisa menekan tombol Finish untuk tahapan ini (yang akan dilanjutkan dengan proses konfigurasi).



Gambar 1.7 MySQL Server 5.1 Setup Wizard (3)

Langka 8

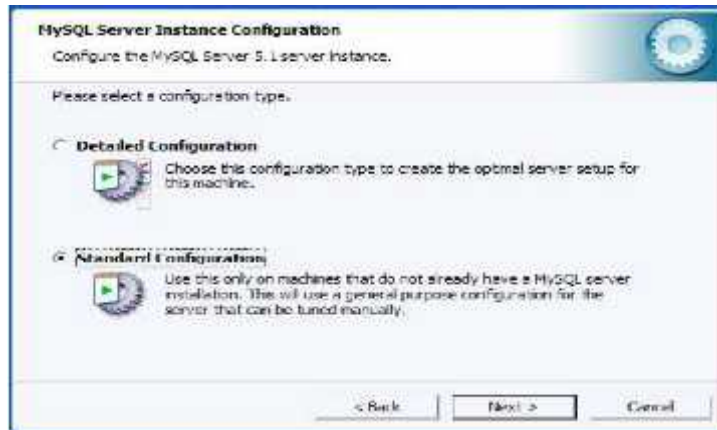
Sudah siap untuk mengkonfigurasi MySQL server? Silakan klik tombol Next...



Gambar 1.8 MySQL Server Instance Configuration Wizard (1)

Langka 9

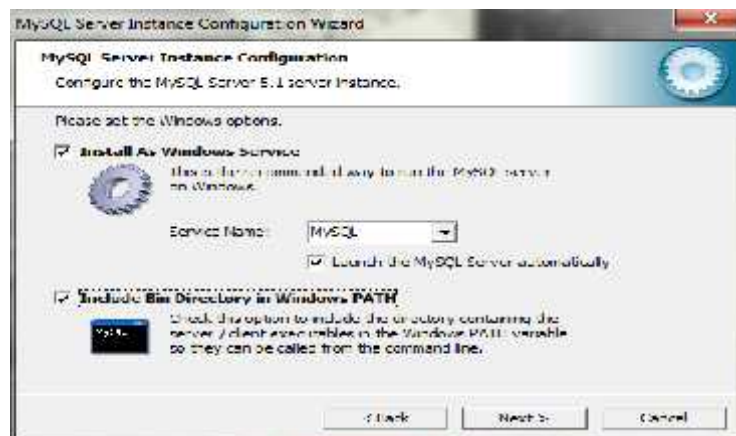
Ada dua pilihan konfigurasi, Detailed Configuration dan Standard Configuration. Kita pilih saja Standard Configuration, karena mudah untuk diselesaikan. Klik pada tombol Next > untuk melanjutkan.



Gambar 1.9 MySQL Server Instance Configuration Wizard (2)

Langka 10

Tampilan berikutnya, disarankan untuk mengaktifkan pilihan Install as Windows Service dan juga Launch the MySQL Server automatically. Dengan pilihan ini maka setiap komputer Anda dinyalakan, secara otomatis program MySQL server akan dijalankan. Begitupun sebaiknya aktifkan pilihan Include Bin directory in Windows Path. Program-program MySQL biasanya disimpan di dalam directory C:\Program Files\MySQL\MySQL Server 5.0\Bin. Dengan mengaktifkan pilihan ini, maka Anda dapat menjalankan atau memanggil program MySQL langsung dari DOS/Command Prompt.

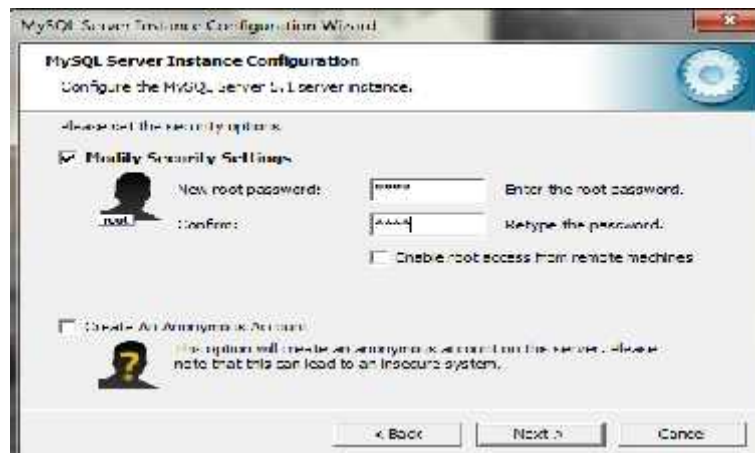


Gambar 1.10 MySQL Server Instance Configuration Wizard (3)

Langka 11

Tampilan berikutnya, mengenai sistem keamanan server MySQL. Sebaiknya membuat password khusus sebagai Root, dan tidak memberikan peluang kepada orang lain untuk memasuki sistem tanpa password.

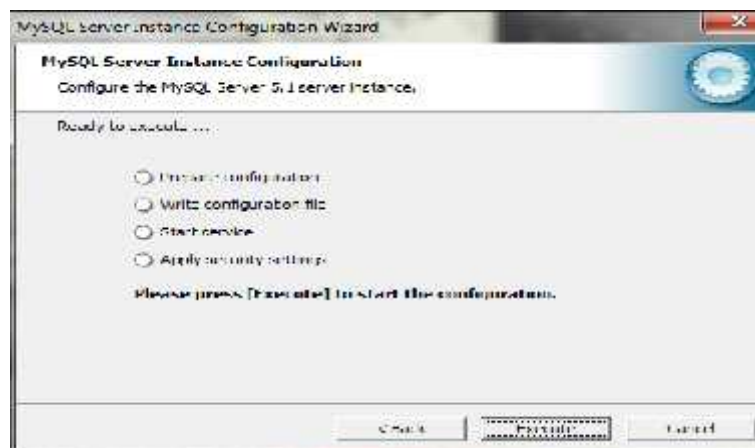
Maka aktifkan pilihan Modify Security Setting dan masukkan password Root dengan seksama. Tetapi, matikan pilihan Create An Anonymous Account. Dengan demikian tidak sembarangan orang dapat masuk menggunakan MySQL server. Satu hal lagi, disarankan mematikan pilihan Enable Root access from remote machines. Ini untuk mencegah celah-celah yang bisa dimasuki oleh orang-orang yang tidak bertanggung jawab menyelip ke dalam sistem. Lanjutkan dengan menekan tombol Next.



Gambar 1.11 MySQL Server Instance Configuration Wizard(4)

Langka 12

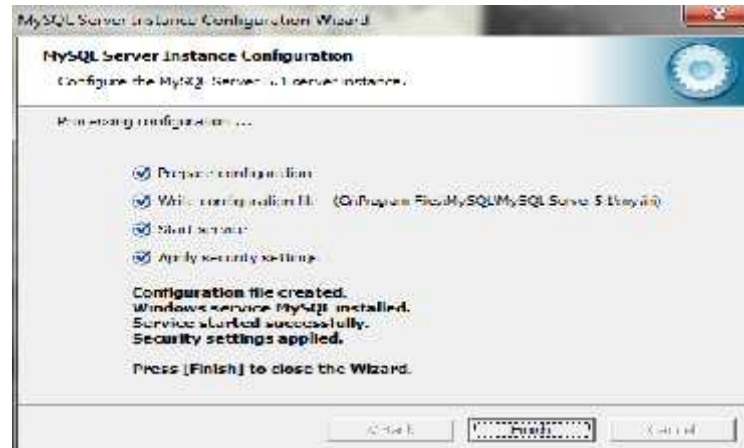
Bila telah yakin untuk melanjutkan, klik pada tombol Execute. Dan bisa santai sejenak sambil menunggu proses setting selesai.



Gambar 1.12 MySQL Server Instance Configuration Wizard(5)

Langka 13

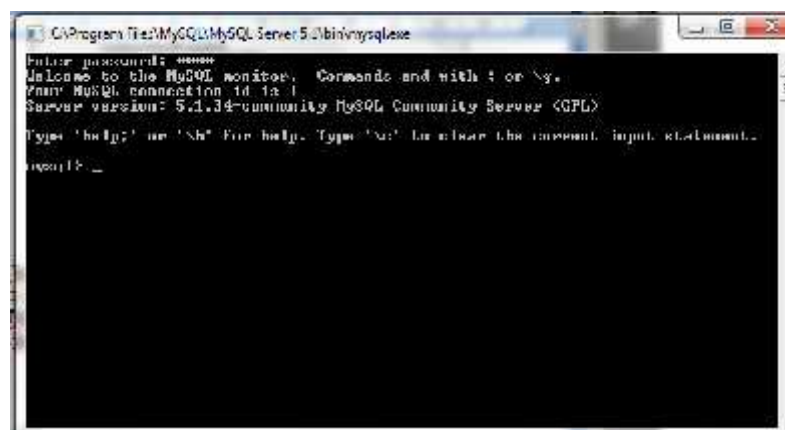
Bila tidak ada kendala apapun, maka selesailah keseluruhan proses instalasi dan setting pada program MySQL 5.1.x ini. Dan klik pada tombol Finish untuk menuntaskan proses ini.



Gambar 1.13 MySQL Server Instance Configuration Wizard(6)

Langka 14

Melakukan uji coba untuk mengakses MySQL dari DOS Prompt/Command Line. Jalankan program MySQL Server melalui menu utama Windows: Start -> Programs -> MySQL -> MySQL Server 5.0 -> MySQL Command Line Client. Kemudian ketikkan password yang telah dibuat pada saat proses instalasi:



Gambar 1.14 MySQL exe


Langka 15

Mencoba dengan perintah sederhana lainnya seperti SHOW DATABASES; untuk menampilkan semua database yang terdapat di MySQL.

```
mysql> show databases;
+-----+
| Database |
+-----+
| information_schena |
| mysql |
| test |
| welnus |
+-----+
4 rows in set (0.00 sec)
```

Gambar 1.15 Menggunakan Perintah Show Databases

Langka 16

Untuk keluar dari sistem MySQL, ketikkan perintah “\q;” atau klik pada tombol close 

3.5. Pembuatan Database

Cara Membuat database di MySQL

Sudah ada 3 buah database di dalam sistem MySQL. Sekarang penulis akan membuat sebuah database Tentang PegarsipanData Personil Ditlantas Polda Sulut.

Begini cara membuatnya.

Mengunakan perintah "CREATE DATABASE" untuk membuat sebuah database.

```
mysql> create database welmus2;
Query OK, 1 row affected (0.02 sec)
```

Melihat hasil dari perintah di atas dengan megetik printah “Show DataBases”


```
mysql> show databases;
+-----+
| Database |
+-----+
| information_schema |
| Lantasi |
| mysql |
| test |
| welmus |
| welmus1 |
| welmus2 |
+-----+
7 rows in set (0.14 sec)

mysql>
```

Tabel 1.1 Menggunakan perintah "SHOW DATA BASES"

Langka selanjutnya pilih database "welmus2" dan kita buka dengan perintah "USE"

```
mysql> use welmus2 ;
Database change
```

```
mysql> use welmus2;
Database changed
```

Gambar 1.2 Menggunakan perintah "USE"

Untuk melihat apa isi dari sebuah database, kita gunakan perintah "SHOW TABLES".

```
mysql> show tables ;
Empty set (0.00 sec)
```

```
mysql> show tables;
Empty set (0.00 sec)
```

Gambar 1.3 Menggunakan perintah "SHOW TABLES"

Hasil dari perintah SHOW TABLES diatas adalah "Empty Set", yang berarti belum ada tabel apapun di dalam database welmus2.

Penulis akan membuat sebuah tabel baru dengan menggunakan perintah "CREATE TABLE" dan menentukan minimal satu buah field/kolom di dalamnya

```
mysql> create table lantasi
-> (nopeg INT UNSIGNED AUTO_INCREMENT PRIMARY KEY)
```



```

EY,
-> nama VARCHAR(50) NOT NULL)
-> ;
Query OK, 0 rows affected (0.14 sec)

```

```

mysql> create table lantaz
-> (nopeg INT UNSIGNED AUTO_INCREMENT PRIMARY KEY,
-> nama VARCHAR(50) NOT NULL)
-> ;
Query OK, 0 rows affected (0.21 sec)

mysql> show tables;
+-----+
| Tables_in_welms2 |
+-----+
| lantaz           |
+-----+
1 row in set (0.00 sec)

mysql> describe lantaz;

```

Tabel 1.4 Menggunakan Perintah "CREATE"

Penulis membuat sebuah tabel Karyawan dengan 2 buah kolom/field.

Kolom pertama adalah NOPEG dengan jenis data bilangan bulat (INTEgerINTEger), tanpa tanda negatif (UNSIGNED), yang akan bertambah nilainya secara otomatis (AUTO_INCREMENT), kolom NOPEG adalah kolom utama (PRIMARY KEY).

Pada kolom kedua, NAMAakan menampung nama karyawan, dengan jenis data VARIabel CHARacter, lebar datanya dapat menampung maksimal 50 karakter, dan tidak boleh dikosongkan (NOT NULL). Kurang lebih seperti itulah ceritanya..)

Melihat isi database welms2:

```

mysql> create table lantaz
-> (nopeg INT UNSIGNED AUTO_INCREMENT PRIMARY KEY,
-> nama VARCHAR(50) NOT NULL)
-> ;
Query OK, 0 rows affected (0.21 sec)

mysql> show tables;
+-----+
| Tables_in_welms2 |
+-----+
| lantaz           |
+-----+
1 row in set (0.00 sec)

```

Tabel 1.5 isi dari database welms2

Dari hasil perintah di atas, kita lihat bahwa database welms2 telah memiliki sebuah tabel yang bernama lantaz. Selanjutnya melihat apa struktur dari tabel Lantaz tersebut.

Untuk melihat struktur sebuah tabel dapat menggunakan perintah "DESCRIBE" atau bisa juga menggunakan perintah "SHOW COLUMNS FROM".

```
mysql> describe lantas ;
```

```
mysql> describe lantas;
+-----+-----+-----+-----+-----+-----+
| Field | Type          | Null | Key | Default | Extra          |
+-----+-----+-----+-----+-----+-----+
| nopeg | int(10) unsigned | NO   | PRI | NULL    | auto_increment |
| nana  | varchar(50)     | NO   |     | NULL    |                |
+-----+-----+-----+-----+-----+-----+
2 rows in set (0.01 sec)
```

Tabel 1.6 isi dari describe lantas

```
mysql> show columns from lantas;
```

```
mysql> show columns from lantas;
+-----+-----+-----+-----+-----+-----+
| Field | Type          | Null | Key | Default | Extra          |
+-----+-----+-----+-----+-----+-----+
| nopeg | int(10) unsigned | NO   | PRI | NULL    | auto_increment |
| nana  | varchar(50)     | NO   |     | NULL    |                |
+-----+-----+-----+-----+-----+-----+
2 rows in set (0.01 sec)
```

Tabel 1.7 isi dari show columns from lantas

Tidak ada perbedaan hasil dari dua perintah di atas bedah kata perintah tapi hasilnya sama.

Mengubah Struktur Sebuah Tabel

Penulis akan mengubah struktur tabel yang pernah dibuat sebelumnya. Pengubahan struktur bisa dalam hal penambahan kolom (ADD), pengubahan lebar dan jenis kolom (MODIFY), atau bisa saja penghapusan kolom dan indeks (DROP), penggantian nama kolom (CHANGE), penggantian nama tabel (RENAME), dan sebagainya. Apa pun juga yang anda lakukan pada kolom tersebut tentu akan mempunyai dampak langsung pada data-data yang sudah ada. Sekarang penulis akan menambahkan beberapa kolom baru, yaitu kolom jenis kelamin, pangkat, nrp dan tglahir pada tabel lantas.

Perintah untuk mengubah struktur tabel adalah "ALTER TABLE". Dengan cara.

```
mysql> ALTER TABLE karyawan
-> ADD jenkelamin CHAR(2) NOT NULL,
-> ADD kota VARCHAR(25) NOT NULL,
```

-> ADD kodepos CHAR(5) NOT NULL,
 -> ADD tgllahir DATE
 ->;
 Query OK, 0 rows affected (0.20 sec)
 Records: 0 Duplicates: 0 Warnings: 0

```
mysql> ALTER TABLE lantasi
  -> ADD jenkelamin CHAR(2) NOT NULL,
  -> ADD pangkat VARCHAR(25) NOT NULL,
  -> ADD nrp CHAR(8) NOT NULL,
  -> ADD tgllahir DATE
  ->;
Query OK, 0 rows affected (0.56 sec)
Records: 0 Duplicates: 0 Warnings: 0

mysql> describe lantasi;
+-----+-----+-----+-----+-----+-----+
| Field          | Type          | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| nopeg          | int(10) unsigned | NO   | PRI | NULL    | auto_incre |
| nama           | varchar(50)      | NO   |     | NULL    |             |
| jenkelamin     | char(2)          | NO   |     | NULL    |             |
| pangkat        | varchar(25)      | NO   |     | NULL    |             |
| nrp            | char(8)          | NO   |     | NULL    |             |
| tgllahir       | date             | YES  |     | NULL    |             |
+-----+-----+-----+-----+-----+-----+
6 rows in set (0.01 sec)
```

Gambar 1.8 isi dari perintah ALTER TABLE

Jenis kelamin hanya membutuhkan lebar data 1, oleh karena itu penulis akan mengubah lebar data pada jenis kelamin tersebut dengan perintah MODIFY,

```
mysql> ALTER TABLE lantasi MODIFY jenkelamin Char(1);
Query OK, 0 rows affected (0.25 sec)
Records: 0 Duplicates: 0 Warnings: 0
```

```
mysql> ALTER TABLE lantasi MODIFY jenkelamin Char(1);
Query OK, 0 rows affected (0.54 sec)
Records: 0 Duplicates: 0 Warnings: 0
```

Gambar 1.9 menggunakan perintah MODIFY

Ada juga cara untuk mengubah nama kolom nopeg diubah saja menjadi noid. Begitupun dengan nama kolom jenkelamin, kita ubah namanya menjadi jenkelsaja. Dalam perubahan kolom ini sebaiknya 'sifat-sifat' kolom yang asli tetap ditulis ulang. Misal bila kolom nopeg memiliki sifat 'auto_increment', maka selama sifat itu tetap dipertahankan, maka dia (auto_increment) harus

ditulis ulang. Begini caranya... Mengubah kolom nopeg menjadi noid, tanpa mengubah jenis datanya (tetap INT(10), dan tetap auto_increment):

Mengubah kolom nopeg menjadi noid

```
mysql> alter table karyawan
->change nopeg noid int(10) auto_increment
->;
Query OK, 0 rows affected (0.16 sec)
Records: 0 Duplicates: 0 Warnings: 0
```

Mengubah kolom jenkelamin menjadi jenkel,

```
mysql> alter table karyawan
->change jenkelamin jenkel char(1);
Query OK, 0 rows affected (0.24 sec)
Records: 0 Duplicates: 0 Warnings: 0
```

```
mysql> alter table lantas
->change nopeg noid int(10) auto_increment
->;
Query OK, 0 rows affected (0.61 sec)
Records: 0 Duplicates: 0 Warnings: 0

mysql> alter table lantas
->change jenkelamin jenkel char(1);
Query OK, 0 rows affected (0.12 sec)
Records: 0 Duplicates: 0 Warnings: 0

mysql> describe lantas;
+----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+----+-----+-----+-----+-----+-----+
| noid | int(10) | NO | PRI | NULL | auto_increment |
| nama | varchar(50) | NO | | NULL | |
| jenkel | char(1) | YES | | NULL | |
| pangkat | varchar(25) | NO | | NULL | |
| nrp | char(8) | NO | | NULL | |
| tgllahir | date | YES | | NULL | |
+----+-----+-----+-----+-----+-----+
6 rows in set (0.01 sec)
```

Tabel 1.10 yang telah di rubah nama kolom menjadi noid dan jenkel.

Mengisi data ke dalam table

Cara mengisi data lantas ke dalam tabel. Perintah yang digunakan adalah "INSERT INTO". Cara yang pertama sebagai berikut:

```
mysql> insert into karyawan
->(nama, jenkel, pangkat, nrp, tgllahir)
->values
->("Jeri repi", "L", "Briptu","23109867","1980-11-03")
->;
```

```
mysql> insert into lantasi
-> (nama, jenkel, pangkat, nrp, tgllahir)
-> values
-> ('jeri repi', 'L', 'pripta', '23109067', '1980-11-03')
-> ;
Query OK, 1 row affected (0.14 sec)
```

Gambar 1.11 mengisi data dalam table dengan perintah INSERT INTO.

Pemasukkan data yang berjenis karakter, selalu diapit dengan tanda kutip ganda ("). Bisa juga digunakan tanda kutip tunggal ('). Tetapi jangan dicampur dengan tanda kutip ganda dan tanda kutip tunggal, misal: "Ahmad Zobari". Perhatikan juga pada penulisan tanggal lahir, menggunakan format "tahun-bulan-tanggal". Memang sedikit susah. Tapi begitulah memang standar MySQL untuk format penulisan tanggal. Mengapa kita tidak memasukkan data untuk kolom "noid" Ini karena sifat kolom noid yang auto_increment, sehingga dia akan secara otomatis berisi dengan angka 1, dan terus bertambah 1, seiring dengan penambahan data.

Bisa juga memasukkan data dengan cara yang lain:

```
mysql> insert into karyawan
-> set nama="Dana Taluke",
-> jenkel="P",
-> kota="Bripka",
-> kodepos="44567643",
-> tgllahir="1995-01-020"
-> ;
```

```
mysql> insert into lantasi
-> set nama="Dana Taluke",
-> Jenkel="P",
-> Pangkat="Bripka",
-> nrp="44567643",
-> tgllahir="1995-01-20"
-> ;
Query OK, 1 row affected (0.11 sec)
```

Gambar 1.12 cara kedua memasukan data dngan perintah insert into.

Melihat data pada table yang telah dimasukan menggunakan perintah "SELECT".

```
mysql> select * from lantasi;
+----+-----+-----+-----+-----+-----+
| noid | nama          | jenkeli | pangkat | nrip      | tglahir  |
+----+-----+-----+-----+-----+-----+
| 1    | isnael jacobus | L       | konpol  | 41109067 | 1977-10-02 |
| 2    | jeri repi      | L       | priptu   | 23109067 | 1980-11-03 |
| 3    | Vivinhoo Taluke | P       | Bripka   | 88997722 | 1993-12-19 |
| 4    | Dana Taluke    | P       | Bripka   | 44567643 | 1995-01-20 |
+----+-----+-----+-----+-----+-----+
4 rows in set (0.07 sec)

mysql>
```

Tabel 1.13 melihat data pada table menggunakan perintah SELECT.

Perintah di atas menampilkan seluruh data yang ada di dalam tabel lantasi, karena menggunakan tanda asterik "*" di dalam perintah SELECT.

Bisa juga hanya mau menampilkan datanama dan jenis kelamin saja dengan perintah SELECT.

```
mysql> select nama, jenkeli from lantasi
-> ;
+-----+-----+
| nama          | jenkeli |
+-----+-----+
| isnael jacobus | L       |
| jeri repi      | L       |
| Vivinhoo Taluke | P       |
| Dana Taluke    | P       |
+-----+-----+
4 rows in set (0.00 sec)

mysql> select nama, jenkeli from lantasi
-> where jenkeli="P"
-> ;
```

Tabel 1.14 hanya menampilkan nama dan jenkeli saja.

Bisa juga hanya mau menampilkan data-data lantasi yang berjenis kelamin perempuan saja dengan perintah cukup dengan menggunakan perintah WHERE saja pada SELECT.

```
mysql> select nama, jenkeli from karyawan
-> where jenkeli="P"
-> ;
```

```
mysql> select nama, jenkel from lantasi
-> where jenkel="p"
-> ;
```

nama	jenkel
Wiwinhoo Taluke	P
Dana Taluke	P

```
2 rows in set (0.04 sec)
```

Tabel 1.15 hanya menampilkan data berjenis kelamin perempuan.

Penulis akan menambahkan sebuah kolom field lagi supaya lengkap dalam pembuatan arsip data polda sulut, yaitu kolom gaji. Kolom Gaji merupakan kolom numeric yang menampung data gaji pokok karyawan perbulannya. Jadi, perlukan adalah jenis data INTeget dengan lebar data 12 digit. Penerapannya sebagai berikut dengan menggunakan perintah ALTER.

```
mysql> alter table karyawan
-> ADD gaji INT(12) NOT NULL default 0
-> ;
```

```
mysql> alter table lantasi
-> ADD gaji INT(12) NOT NULL default 0
-> ;
Query OK, 4 rows affected (0.86 sec)
Records: 4 Duplicates: 0 Warnings: 0
```

Gambar 1.16 menambahkan sebuah kolom Field (Gaji)

Periksa struktul table dulu:

```
mysql> describe lantasi;
```

Field	Type	Null	Key	Default	Extra
noid	int(10)	NO	PRI	NULL	auto_increment
nama	varchar(50)	NO		NULL	
jenkel	char(1)	YES		NULL	
pangkat	varchar(25)	NO		NULL	
nep	char(8)	NO		NULL	
tgllahir	date	YES		NULL	
gaji	int(12)	NO		0	

```
7 rows in set (0.01 sec)
```

Tabel 1.17 kolom gaji telah di tambakan kedalam table Lantasi

Kolom gaji sudah ditambahkan ke dalam tabel Lantasi. Sekarang penulis menambahkan data gaji kepada tiap-tiap karyawan yang ada. Untuk memudahkan, tampilkan dulu semua data yang ada di tabel Lantasi:

Menggunakan perintah select.
mysql> select * from karyawan ;

```
mysql> select * from lantas;
```

noid	nama	jenkel	pangkat	nrp	tgllahir	gaji
1	isnael jacobus	L	kompol	41109067	1977-10-02	0
2	jeri repi	L	pripta	23109067	1980-11-03	0
3	Wiwinhoo Taluke	P	Bripka	88997722	1993-12-19	0
4	Dana Taluke	P	Bripka	44567643	1995-01-20	0

```
4 rows in set (0.00 sec)
```

Tabel 1.18 Tampilan semua data yang ada di table Lantas.

Meng-Update Data Pada Table

Masukkan data gaji masing-masing anggota lantas dengan menggunakan perintah UPDATE. Mulai dari Ahmad Zobari, dengan noid=1:

```
mysql> update karyawan  
->set gaji=1000000  
->where noid=1 ;  
Query OK, 1 row affected (0.09 sec)  
Rows matched: 1 Changed: 1 Warnings: 0
```

Periksa hasilnya

Menggunakan perintah SELCET *

```
mysql> select * from lantas  
->where noid=1 ;
```

```
mysql> select * from lantas  
-> where noid=1;
```

noid	nama	jenkel	pangkat	nrp	tgllahir	gaji
1	isnael jacobus	L	kompol	41109067	1977-10-02	1000000

```
1 row in set (0.00 sec)
```

Tabell.19 Data gaji telah dimasukan kedalam table Lantas

Lanjutkan dengan anggota lantas lainnya, seperti Jeri Repidengan noid=2, wiwinhoo taluke dengan noid=3, dan seterusnya. Sayangnya, perintah ini

hanya bisa dilakukan satu per satu. Jadi, harus sabar menjalankan perintah di bawah ini yaa ...:

```
mysql> update lantasi
->set gaji=1250000 where noid=2 ;
Query OK, 1 row affected (0.39 sec)
Rows matched: 1 Changed: 1 Warnings: 0
```

```
mysql> update karyawan
->set gaji=1500000 where noid=3 ;
Query OK, 1 row affected (0.03 sec)
Rows matched: 1 Changed: 1 Warnings: 0
```

```
mysql> update karyawan
->set gaji=2000000 where noid=4 ;
Query OK, 1 row affected (0.03 sec)
Rows matched: 1 Changed: 1 Warnings: 0
```

```
mysql> update lantasi
-> set gaji=1250000 where noid=2;
Query OK, 1 row affected (0.19 sec)
Rows matched: 1 Changed: 1 Warnings: 0

mysql> update lantasi
-> set gaji=1500000 where noid=3;
Query OK, 1 row affected (0.14 sec)
Rows matched: 1 Changed: 1 Warnings: 0

mysql> update lantasi
-> set gaji=2000000 where noid=4;
Query OK, 1 row affected (0.11 sec)
Rows matched: 1 Changed: 1 Warnings: 0

mysql> select * from lantasi;
+----+-----+-----+-----+-----+-----+-----+
| noid | nama          | jenkel | pangkat | nip      | tanggalahir | gaji |
+----+-----+-----+-----+-----+-----+-----+
| 1    | Idrisael Jusubus | L      | Koprol  | 41107967 | 1977-10-02   | 1000000 |
| 2    | Jeri Reji      | L      | prapta  | 23107967 | 1980-11-03   | 1250000 |
| 3    | Wirinhu Taluke | P      | Brpka   | 88997722 | 1993-12-19   | 1500000 |
| 4    | Dana Taluke    | P      | Brpka   | 44507040 | 1995-01-20   | 2000000 |
+----+-----+-----+-----+-----+-----+-----+
4 rows in set (0.00 sec)

mysql>
```

Tabel 1.20.Data gaji yang telah di update

```
mysql> select * from lantasi;
```

noid	nama	jenkel	pangkat	nrp	tgllahir	gaji
1	isnael jacobus	L	kompul	41109067	1977-10-02	1000000
2	jeri repi	L	priptu	23109067	1980-11-03	1250000
3	Wiwinho Taluke	P	Bripka	88997722	1993-12-19	1500000
4	Dana Taluke	P	Bripka	44567643	1995-01-20	2000000
5	Welnus Ianera	L	AKBP	66999688	1995-01-20	1200000
6	Hani Tahapari	L	Kompul	66349656	1966-04-25	3000000
7	Falen Lobiua	NULL	iptu	66345632	1995-06-04	1200000
8	rial nararu	L	ipda	57845632	1994-07-05	3500000
9	eko lolorie	L	Kompul	57847781	1985-02-26	3200000
10	Roni Tamboto	L	lptu	45776789	1980-05-24	2100000
11	skurman wadui	NULL	lpda	45771129	1980-02-23	1500000
12	linda nakal	NULL	kompul	34971129	1960-02-23	1700000
13	doni pamungkas	NULL	iptu	34873429	1995-04-09	2300000
14	tasya taluningan	NULL	bripda	67273429	1992-12-06	2700000
15	sapri	NULL	ipda	67276669	1955-12-06	2500000
16	linda lobiua	P	kompul	67276678	1951-12-06	3700000

BAB IV

KESIMPULAN DAN SARAN

4.1. Bagian Akhir

a. Kesimpulan

Berdasarkan judul dari pembuatan studi kasus diatas yaitu pengarsipan personil Ditlantas Polda Sulut menggunakan Mysql,maka diambil kesimpulan seperti di bawa ini.

- Dalam pembuatan sistem database menggunakan Myqsl haruslah banyak menggunakan perintah-perinta/kode-kode yang bermacam-macam dan cara mengisi database di mysql masi pakai cara manual untuk itu di butukan kesabaran,ketelitian dan pemahaman tentang pebuatan databade di MsqL.

b.Saran

Saran yang dapat di berikan dalam pembuatan studi kasus ini adalah.

- Untuk pembuatan studi kasus ini kedepanya perlu di tinjau lebih jau untuk mengkoneksikan Mysql dang Php supaya lebih baik lagi dalam pembuatan desainya.

c. Daftar Pustaka

- <https://shirotholmustaqim.files.wordpress.com/2010/02/modul-mysql-revisi.pdf>
- <http://www.mandalamaya.com/pengertian-database/>
- <http://www.etunas.com/web/pengertian-mysql.htm>
- http://sudibyodibyoblogspot.blogspot.com/2012/03/kelebihan-dan-kekurangan-mysql_07.html
- <http://www.jasawehtasik.blogspot.com/2013/07/tujuan-dan-manfaat-basis-data.html>