



BUKU AJAR

Manajemen Database Client Server

Gabriel Ardi Hutagalung, S.T., M.Kom
Hikmah Adwin Adam, S.Kom., M.Kom
Yuyun Yusnida Lase, S.Kom., M.Kom
Santi Prayudani, S.Kom., M.Kom
Arif Ridho Lubis, B.IT., M.Sc.IT



BUKU AJAR MANAJEMEN DATABASE CLIENT SERVER

GABRIEL ARDI HUTAGALUNG, S.T., M.Kom.
HIKMAH ADWIN ADAM, S.Kom., M.Kom.
YUYUN YUSNIDA LASE, S.Kom., M.Kom.
SANTI PRAYUDANI, S.Kom., M.Kom.
ARIF RIDHO LUBIS, B.IT., M.Sc.IT.



BUKU AJAR
MANAJEMEN DATABASE CLIENT SERVER

© 2023 All rights reserved. No part of this book may be reprinted or reproduced or utilised in any form or by any electronic, mechanical, or other means, now known or hereafter invented, including photocopying and recording, or in any information storage or retrieval system, without permission in writing from the writer

Penulis:

Gabriel Ardi Hutagalung, S.T., M.Kom.

Hikmah Adwin Adam, S.Kom., M.Kom.

Yuyun Yusnida Lase, S.Kom., M.Kom.

Santi Prayudani, S.Kom., M.Kom.

Arif Ridho Lubis, B.IT., M.Sc.IT.

Desain Sampul:

Rezha Destiadi, S.Sn., M.Ds.

Tata Letak:

Isni Khairina, S.Pd., M.Pd.

Cetakan I: April 2023

viii, 73 halaman, 25 cm

ISBN: 978-623-5963-53-2

Diterbitkan oleh

Penerbit PolmedPress

(Member of IKAPI)

Politeknik Negeri Medan

Gedung PUML, Almamater No.1

Email: polmedpress@polmed.ac.id

PRAKATA

Manajemen Database Client Server merupakan Mata Kuliah wajib di program studi Teknologi Rekayasa Perangkat Lunak dan Manajemen Informatika, dimana pada mata kuliah ini, mahasiswa akan mendapatkan pengetahuan dan keahlian mengenai pengelolaan database. Di dalam buku ini dijelaskan bagaimana query manajemen database itu dilakukan, dimulai dari manajemen struktur database, sampai manajemen isi dan backup dari database secara keseluruhan. maka sebagai pengantar didalam buku ini akan disajikan pengenalan dengan konsep database. Selanjutnya penulis juga menyampaikan dasar-dasar DDL, DML dengan mengenalkan perintah dasar CRUD sampai ke tahap manipulasi data sampai ke tahapan backup.

Semoga dengan adanya buku ajar ini, mahasiswa yang membutuhkan refrensi pembelajaran dapat terbantu dalam mata kuliah Manajemen Database Client Server.

Gabriel Ardi Hutagalung/Hikmah Adwin Adam/Yuyun Yusnida
Lase/Santi Prayudani/Arif Ridho Lubis/

DAFTAR ISI

Halaman

PRAKATA	III
DAFTAR ISI	IV
DAFTAR GAMBAR.....	VI
INTRODUKSI.....	VII
BAB I DATABASE KONSEP	1
BAB II MODEL DATA	5
MODEL DATA	5
MODEL HUBUNGAN ENTITAS	5
MODEL RELASIONAL	7
BAB III MANAJEMEN DATABASE	8
SISTEM MANAJEMEN DATABASE	8
DBMS MENGELOLA DATA	8
DBMS CLOUD	12
KEUNTUNGAN MENGGUNAKAN DBMS.....	13
BAB IV MYSQL WORKBENCH	15
CARA INSTALL MYSQL WORKBENCH	17
CARA MENGGUNAKAN MYSQL WORKBENCH	23
MYSQL WORKBENCH ADMINISTRATION TOOL	26
BAB V DATA DEFENITION LANGUAGE & DATA MANIPULATION LANGUAGE.....	28
DATA DEFENITION LANGUAGE	28
CREATE.....	28
READ	29
UPDATE.....	30
DELETE	32
DATA DATA MANIPULATION LANGUAGE	33
CREATE.....	33
READ	35
UPDATE	35
DELETE	36
BAB VI WHERE CLAUSE PADA SQL.....	37
TEXT FIELDS VS. NUMERIC FIELDS	37
MYSQL AND, OR AND NOT OPERATORS.....	41
SORTING DATA.....	43
LIMIT CLAUSE	44
MIN() AND MAX() FUNCTIONS.....	46

COUNT(), AVG() AND SUM() FUNCTIONS.....	46
BAB VII JOIN PADA TABEL	48
INNER JOIN	48
LEFT JOIN	50
RIGHT JOIN.....	50
BAB VIII PROCEDURE PADA MYSQL.....	52
FITUR STORE PROCEDURE	52
BAGAIMANA CARA MEMBUAT PROSEDUR?	53
BAB IX SERANGAN PADA MYSQL	55
SQL INJECTION	55
CROSS SITE SCRIPTING (XSS) – CLIENT HIJACKING	58
CVE-2012-5614 – DENIAL OF SERVICE	61
CVE-2016-6663 – RACE CONDITION	62
CVE-2012-5615 — REMOTE PREAUTH USER ENUMERATION	64
BAB X PENCADANGAN DATA PADA MYSQL.....	65
LOGICAL BACKUP.....	65
PHYSICAL BACKUP	66
CONSISTENT	66
PERINTAH MYSQLDUMP	67
MYSQL WORKBENCH TO BACK UP A DATABASE	68
DAFTAR PUSTAKA	71
BIOGRAFI PENULIS	72

DAFTAR GAMBAR

Gambar 2. 1 Konsep ERD	6
Gambar 4. 1 Logo MySQL	17
Gambar 4. 2 Tampilan Download MySQL Work Bench	18
Gambar 4. 3 MySQL Community Download.....	18
Gambar 4. 4 MySQL Installer	19
Gambar 4. 5 MySQL Installer Exe.....	19
Gambar 4. 6 MySQL Installer.....	20
Gambar 4. 7 MySQL Installer	21
Gambar 4. 8 MySQL Installer.....	21
Gambar 4. 9 Installation Complete	22
Gambar 4. 10 Wellcome Message	23
Gambar 4. 11 Input Password.....	24
Gambar 4. 12 Input User dan Password	25
Gambar 4. 13 Success Login	25
Gambar 4. 14 Dashboard Workbench	26
Gambar 7. 1 Inner Join Ilustrasi	49
Gambar 7. 2 Ilustrasi Left Join	50
Gambar 7. 3 Ilustrasi Right Join (DQLab 2022)	51
Gambar 10. 1 Backup Database	69

INTRODUKSI

Penggunaan buku ini sebagai bahan pembelajaran mengacu pada Rencana Pembelajaran Semester (RPS) di Politeknik Negeri Medan (Polmed). Sebelum menggunakan buku ajar ini sebaiknya mahasiswa memahami terlebih dahulu hakikat perkuliahan “Manajemen Database Client Server” sebagai berikut.

Deskripsi Singkat Mata Kuliah

Mata kuliah menyajikan pembelajaran tentang Manajemen Database baik dari pengeolahan Query, Backup Database, dan Upload Database ke Server.

Capaian Pembelajaran Mata Kuliah

(Jelaskan kemampuan mahasiswa yang diharapkan)

Mahasiwa mampu memanajemen database, mengelola database, melakukan backup database.

Materi Pembelajaran

Berikut ini materi pembelajaran untuk mata kuliah “Manajemen Database Client Server”:

1. DDL
2. DML
3. Join, Sorting.
4. Backup

Prasyarat Pembelajaran

Mahasiswa telah mengikuti mata kuliah Sistem Basis Data.

Penggunaan Buku Ajar

Mahasiswa telah mengikuti mata kuliah Sistem Basis data dan Mampu melakukan pengolahan CRUD baik dengan DML maupun DDL.

BAB I

DATABASE KONSEP

Database memiliki kumpulan data yang disimpan bersama untuk melayani beberapa aplikasi sebanyak mungkin. Oleh karena itu, basis data sering dianggap sebagai gudang informasi yang diperlukan untuk menjalankan fungsi tertentu dalam perusahaan atau organisasi. Basis data tidak hanya memungkinkan pengguna melakukan pengambilan data tetapi juga melakukan modifikasi terus menerus dari data yang diperlukan untuk pengendalian operasi. Dimungkinkan untuk mencari database untuk mendapatkan jawaban atas pertanyaan atau informasi untuk tujuan perencanaan.

1. TUJUAN DATABASE

Database harus menjadi tempat penyimpanan data yang dibutuhkan untuk pemrosesan data organisasi. Data itu harus akurat, pribadi, dan dilindungi dari kerusakan. Itu harus akurat sehingga beragam aplikasi dengan persyaratan data yang berbeda dapat menggunakan data tersebut. Pemrogram aplikasi yang berbeda dan pengguna akhir yang berbeda memiliki pandangan yang berbeda terhadap data, yang harus diturunkan dari keseluruhan struktur data umum. Metode pencarian dan akses data mereka akan berbeda.

2. KEUNTUNGAN MENGGUNAKAN DATABASE

1. Basis data sangat meminimalkan redundansi data.
2. Basis data dapat mengontrol sebagian besar ketidakkonsistenan data.
3. Berbagi data juga dimungkinkan dengan menggunakan database.
4. Database menegakkan standar.
5. Penggunaan Database dapat menjamin keamanan data.
6. Integritas dapat dikelola dengan menggunakan database.

3. BERBAGAI TINGKAT DATABASE

Database diimplementasikan melalui tiga tingkatan umum. Level-level ini adalah:

- Tingkat Internal atau tingkat Fisik
- Tingkat Konseptual
- Level Eksternal atau Level Tampilan

4. KONSEP INDEPENDENSI DATA

Karena basis data dapat dilihat melalui tiga level abstraksi, setiap perubahan pada level mana pun dapat memengaruhi skema level lainnya. Karena database terus berkembang, maka mungkin akan sering terjadi perubahan sewaktu-waktu. Ini seharusnya tidak mengarah pada desain ulang dan implementasi ulang database. Konsep kemandirian data terbukti bermanfaat dalam jenis konteks seperti itu.

- Independensi data fisik
- Kemandirian data logis

Relasi: Secara umum, relasi adalah tabel, yaitu data disusun dalam baris dan kolom. Suatu relasi memiliki sifat-sifat sebagai berikut:

- Dalam setiap kolom tabel, semua item memiliki jenis yang sama, sedangkan item dalam kolom yang berbeda mungkin tidak memiliki jenis yang sama.
- Untuk baris, setiap kolom harus memiliki nilai atomik, dan juga untuk baris, kolom tidak boleh memiliki lebih dari satu nilai.
- Semua baris dari suatu relasi adalah berbeda.
- Urutan baris dalam suatu hubungan tidak penting.
- Kolom relasi diberi nama yang berbeda, dan pengurutan kolom ini tidak penting.

Tuple: Deretan tabel dalam suatu hubungan umumnya disebut sebagai Tuple.

Atribut: Kolom atau bidang tabel disebut sebagai Atribut.

Derajat: Jumlah atribut dalam suatu relasi menentukan derajat relasi. Sebuah relasi yang memiliki tiga atribut dikatakan memiliki relasi berderajat 3.

Kardinalitas: Jumlah tupel atau baris dalam suatu relasi disebut sebagai kardinalitas.

Pada tanggal 5 November 2007, Google mengumumkan pendirian Open Handset Alliance, sebuah konsorsium dari puluhan perusahaan teknologi dan telepon seluler, termasuk Intel Corporation, Motorola,

Inc., NVIDIA Corporation, Texas Instruments Incorporated, LG Electronics, Inc., Samsung Electronics, Sprint Nextel Corporation, dan T-Mobile (Deutsche Telekom). Konsorsium dibuat untuk mengembangkan dan mempromosikan Android sebagai sistem operasi open source gratis dengan dukungan untuk aplikasi pihak ketiga. Perangkat berbasis Android menggunakan jaringan nirkabel untuk memanfaatkan sepenuhnya fitur seperti pencarian Google sekali sentuh, Google Docs (misalnya, editor kata, spreadsheet), dan Google Earth (perangkat lunak pemetaan satelit). (Encyclopaedia Britannica 2022).

BAB II

MODEL DATA

MODEL DATA

Model data menentukan bagaimana struktur logis dari database dimodelkan. Model Data adalah entitas mendasar untuk memperkenalkan abstraksi dalam DBMS. Model data menentukan bagaimana data terhubung satu sama lain dan bagaimana mereka diproses dan disimpan di dalam sistem. Model data pertama bisa berupa model data datar, di mana semua data yang digunakan disimpan di bidang yang sama. Model data sebelumnya tidak begitu ilmiah, sehingga cenderung menimbulkan banyak duplikasi dan memperbarui anomali.

MODEL HUBUNGAN ENTITAS

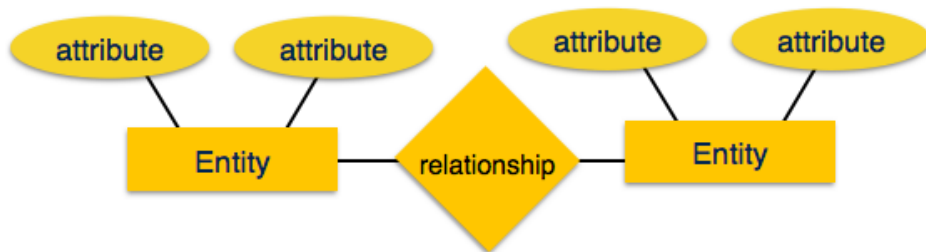
Model Entity-Relationship (ER) didasarkan pada gagasan entitas dunia nyata dan hubungan di antara mereka. Saat merumuskan skenario dunia nyata ke dalam model basis data, Model ER membuat himpunan entitas, himpunan relasi, atribut umum, dan batasan.

Model ER paling baik digunakan untuk desain konseptual database.

Model ER didasarkan pada –

- Entitas dan atributnya.
- Hubungan antar entitas.

Konsep-konsep ini dijelaskan di bawah ini.



Gambar 2. 1 Konsep ERD

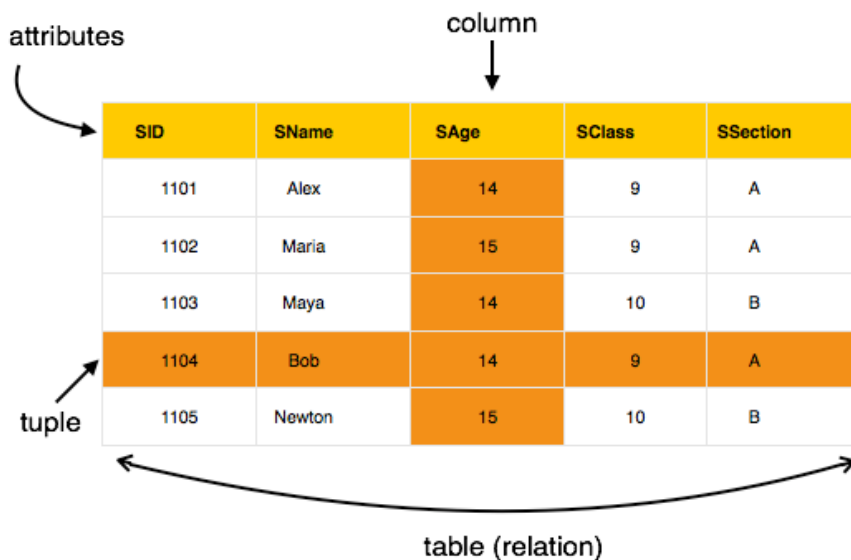
- Entitas – Entitas dalam Model ER adalah entitas dunia nyata yang memiliki properti yang disebut atribut. Setiap atribut didefinisikan oleh kumpulan nilainya yang disebut domain. Misalnya, dalam database sekolah, siswa dianggap sebagai entitas. Siswa memiliki berbagai atribut seperti nama, umur, kelas, dll.
- Hubungan – Asosiasi logis antara entitas disebut hubungan. Hubungan dipetakan dengan entitas dalam berbagai cara. Kardinalitas pemetaan menentukan jumlah asosiasi antara dua entitas.

Pemetaan kardinalitas –

- satu lawan satu
- satu ke banyak
- banyak ke satu
- banyak ke banyak

MODEL RELASIONAL

Model data yang paling populer di DBMS adalah Model Relasional. Ini adalah model yang lebih ilmiah daripada yang lain. Model ini didasarkan pada logika predikat orde pertama dan mendefinisikan tabel sebagai relasi n-ary.



Sorotan utama dari model ini adalah –

- Data disimpan dalam tabel yang disebut relasi.
- Relasi dapat dinormalisasi.
- Dalam relasi yang dinormalisasi, nilai yang disimpan adalah nilai atomik.
- Setiap baris dalam relasi berisi nilai yang unik.
- Setiap kolom dalam relasi berisi nilai dari domain yang sama.

BAB III

MANAJEMEN DATABASE

SISTEM MANAJEMEN DATABASE

Sistem manajemen basis data (DBMS) adalah perangkat lunak sistem untuk membuat dan mengelola basis data. DBMS memungkinkan pengguna akhir untuk membuat, melindungi, membaca, memperbarui, dan menghapus data dalam database. Jenis platform manajemen data yang paling umum, DBMS pada dasarnya berfungsi sebagai antarmuka antara database dan pengguna atau program aplikasi, memastikan bahwa data diatur secara konsisten dan tetap mudah diakses.

DBMS MENGELOLA DATA

Mesin basis data memungkinkan data diakses, dikunci, dan dimodifikasi; dan skema basis data mendefinisikan struktur logis basis data. Ketiga elemen dasar ini membantu menyediakan konkurensi, keamanan, integritas data, dan prosedur administrasi data yang seragam. DBMS mendukung banyak tugas administrasi database tipikal, termasuk manajemen perubahan, pemantauan dan penyetelan kinerja, keamanan, serta pencadangan dan pemulihan. Sebagian besar sistem manajemen database juga bertanggung jawab atas rollback dan restart otomatis serta logging dan audit aktivitas di

database dan aplikasi yang mengaksesnya. DBMS menyediakan tampilan data terpusat yang dapat diakses oleh banyak pengguna dari berbagai lokasi secara terkendali. DBMS dapat membatasi data apa yang dilihat pengguna akhir dan bagaimana mereka melihat data, menyediakan banyak tampilan dari satu skema database. Pengguna akhir dan program perangkat lunak bebas dari keharusan untuk memahami di mana data berada secara fisik atau pada jenis media penyimpanan apa karena DBMS menangani semua permintaan.

DBMS dapat menawarkan independensi data logis dan fisik untuk melindungi pengguna dan aplikasi dari keharusan mengetahui di mana data disimpan atau dari kekhawatiran tentang perubahan pada struktur fisik data. Selama program menggunakan antarmuka pemrograman aplikasi (API) untuk database yang disediakan DBMS, pengembang tidak perlu memodifikasi program hanya karena perubahan telah dilakukan pada database. Dalam sistem manajemen basis data relasional (RDBMS) - jenis DBMS yang paling banyak digunakan - API adalah SQL, bahasa pemrograman standar untuk mendefinisikan, melindungi, dan mengakses data.

KOMPONEN DBMS

DBMS adalah perangkat lunak sistem canggih yang terdiri dari beberapa komponen terintegrasi yang memberikan lingkungan yang konsisten dan terkelola untuk membuat, mengakses, dan memodifikasi data dalam basis data. Komponen-komponen ini termasuk yang berikut:

- Mesin penyimpanan. Elemen dasar DBMS ini digunakan untuk menyimpan data. DBMS harus berinteraksi dengan sistem file pada tingkat sistem operasi (OS) untuk menyimpan data. Itu dapat menggunakan komponen tambahan untuk menyimpan data atau antarmuka dengan data aktual di tingkat sistem file.
- Katalog metadata. Kadang-kadang disebut katalog sistem atau kamus database, katalog metadata berfungsi sebagai tempat penyimpanan untuk semua objek database yang telah dibuat. Ketika database dan objek lain dibuat, DBMS secara otomatis mendaftarkan informasi tentangnya di katalog metadata. DBMS menggunakan katalog ini untuk memverifikasi permintaan pengguna akan data, dan pengguna dapat menanyakan katalog untuk informasi tentang struktur basis data yang ada di DBMS. Katalog metadata dapat menyertakan informasi tentang objek basis data, skema, program, keamanan, kinerja, komunikasi, dan detail lingkungan lainnya tentang basis data yang dikelolanya.
- Bahasa akses basis data. DBMS juga harus menyediakan API untuk mengakses data, biasanya dalam bentuk bahasa akses basis data untuk mengakses dan memodifikasi data, tetapi juga dapat digunakan untuk membuat objek basis data dan mengamankan serta mengotorisasi akses ke data. SQL adalah contoh bahasa akses basis data dan mencakup beberapa set perintah, termasuk Bahasa Kontrol Data untuk mengotorisasi akses data, Bahasa Definisi Data untuk

mendefinisikan struktur basis data, dan Bahasa Manipulasi Data untuk membaca dan memodifikasi data.

- Mesin pengoptimalan. DBMS juga dapat menyediakan mesin pengoptimalan, yang digunakan untuk mengurai permintaan bahasa akses basis data dan mengubahnya menjadi perintah yang dapat ditindaklanjuti untuk mengakses dan memodifikasi data.
- Permintaan prosesor. Setelah kueri dioptimalkan, DBMS harus menyediakan sarana untuk menjalankan kueri dan mengembalikan hasilnya.
- Kunci manajer. Komponen penting dari DBMS ini mengelola akses bersamaan ke data yang sama. Kunci diperlukan untuk memastikan banyak pengguna tidak mencoba mengubah data yang sama secara bersamaan.
- Manajer log. DBMS mencatat semua perubahan yang dilakukan pada data yang dikelola oleh DBMS. Catatan perubahan dikenal sebagai log, dan komponen pengelola log dari DBMS digunakan untuk memastikan bahwa catatan log dibuat secara efisien dan akurat. DBMS menggunakan manajer log selama shutdown dan startup untuk memastikan integritas data, dan antarmuka dengan utilitas database untuk membuat cadangan dan menjalankan pemulihan.
- Utilitas data. DBMS juga menyediakan seperangkat utilitas untuk mengelola dan mengendalikan aktivitas basis data. Contoh utilitas basis data termasuk reorganisasi, runstats, cadangan dan salin, pulihkan, pemeriksaan integritas, muat data dan bongkar data.

Sistem manajemen basis data berbentuk kolom menyimpan data dalam tabel yang berfokus pada kolom, bukan baris, sehingga menghasilkan akses data yang lebih efisien saat hanya subkumpulan kolom yang diperlukan. Ini sangat cocok untuk gudang data yang memiliki sejumlah besar item data serupa. Produk database kolumnar yang populer meliputi Snowflake dan Amazon Redshift.

DBMS Multimodel. Sistem ini mendukung lebih dari satu model database. Pengguna dapat memilih model yang paling sesuai untuk kebutuhan aplikasi mereka tanpa harus beralih ke DBMS yang berbeda. Misalnya, IBM Db2 adalah DBMS relasional, tetapi juga menawarkan opsi kolumnar. Banyak dari sistem basis data paling populer memenuhi syarat sebagai multimodel melalui add-on, termasuk Oracle, PostgreSQL, dan MongoDB. Produk lain, seperti Azure Cosmos DB dan MarkLogic, dikembangkan secara khusus sebagai database multimodel.

DBMS CLOUD

Dibangun dan diakses melalui cloud, DBMS dapat berupa tipe apa saja (relasional, NoSQL, dll.) dan sistem konvensional yang diterapkan dan dikelola oleh organisasi pengguna atau layanan terkelola yang disediakan oleh vendor database. Layanan cloud populer yang memungkinkan penerapan database cloud termasuk Microsoft Azure, Google Cloud, dan AWS.

KEUNTUNGAN MENGGUNAKAN DBMS

Salah satu keuntungan terbesar menggunakan DBMS adalah memungkinkan pengguna dan pemrogram aplikasi mengakses dan menggunakan data yang sama secara bersamaan sambil mengelola integritas data. Data lebih terlindungi dan terpelihara saat dapat dibagikan menggunakan DBMS daripada membuat iterasi baru dari data yang sama yang disimpan dalam file baru untuk setiap aplikasi baru. DBMS menyediakan pusat penyimpanan data yang dapat diakses oleh banyak pengguna secara terkontrol.

Penyimpanan pusat dan pengelolaan data dalam DBMS menyediakan hal-hal berikut:

- Abstraksi dan independensi data;
- Keamanan data;
- Mekanisme penguncian untuk akses bersamaan;
- Penangan yang efisien untuk menyeimbangkan kebutuhan beberapa aplikasi dengan menggunakan data yang sama;
- Kemampuan untuk pulih dengan cepat dari kerusakan dan kesalahan;
- Kemampuan integritas data yang kuat;
- Pencatatan dan audit kegiatan;
- Akses sederhana menggunakan api standar; Dan
- Prosedur administrasi yang seragam untuk data.

Keuntungan lain dari DBMS adalah bahwa administrator basis data (DBA) dapat menggunakannya untuk menerapkan organisasi yang logis dan terstruktur pada data. DBMS memberikan skala ekonomis untuk memproses data dalam jumlah besar karena dioptimalkan untuk operasi semacam itu.

DBMS juga dapat menyediakan banyak tampilan dari skema basis data tunggal. Tampilan menentukan data apa yang dilihat pengguna dan bagaimana pengguna melihat data tersebut. DBMS menyediakan tingkat abstraksi antara skema konseptual yang mendefinisikan struktur logis dari database dan skema fisik yang menjelaskan file, indeks, dan mekanisme fisik lainnya yang digunakan database. DBMS memungkinkan pengguna untuk memodifikasi sistem dengan lebih mudah ketika kebutuhan bisnis berubah. DBA dapat menambahkan kategori data baru ke database tanpa mengganggu sistem yang ada, dengan demikian mengisolasi aplikasi dari cara struktur dan penyimpanan data.

Namun, DBMS harus melakukan pekerjaan tambahan untuk memberikan keuntungan ini, sehingga menimbulkan biaya tambahan. DBMS akan menggunakan lebih banyak memori dan CPU daripada sistem penyimpanan file sederhana, dan jenis DBMS yang berbeda akan memerlukan jenis dan tingkat sumber daya sistem yang berbeda.

BAB IV

MySQL Workbench

MySQL Workbench adalah perangkat lunak lintas platform yang didukung oleh Oracle, yang memungkinkan Anda melakukan pengembangan dengan Database MySQL. MySQL Workbench adalah sistem manajemen basis data sumber terbuka, cepat, andal, dan sangat skalabel. Untuk menggunakan MySQL WorkBench dibutuhkan MySQL, untuk itu pengguna harus menginstall MySQL Terlebih dahulu.

MySQL Workbench memungkinkan DBA, developer, atau arsitek data untuk merancang, memodelkan, membuat, dan mengelola database secara visual. Ini mencakup semua yang dibutuhkan pemodel data untuk membuat model ER yang kompleks, rekayasa maju dan mundur, dan juga menghadirkan fitur utama untuk melakukan tugas manajemen perubahan dan dokumentasi yang sulit yang biasanya memerlukan banyak waktu dan upaya.

MySQL Workbench menghadirkan alat visual untuk membuat, menjalankan, dan mengoptimalkan kueri SQL. Editor SQL menyediakan penyorotan sintaks warna, pelengkapan otomatis, penggunaan kembali cuplikan SQL, dan riwayat eksekusi SQL. Panel Koneksi Basis Data memungkinkan pengembang mengelola koneksi

basis data standar dengan mudah, termasuk MySQL Fabric. Object Browser menyediakan akses cepat ke skema database dan objek.

MySQL Workbench menyediakan konsol visual untuk mengelola lingkungan MySQL dengan mudah dan mendapatkan visibilitas yang lebih baik ke dalam database. Pengembang dan DBA dapat menggunakan alat visual untuk mengonfigurasi server, mengelola pengguna, melakukan pencadangan dan pemulihan, memeriksa data audit, dan melihat kesehatan database.

MySQL Workbench menyediakan seperangkat alat untuk meningkatkan kinerja aplikasi MySQL. DBA dapat dengan cepat melihat indikator kinerja utama menggunakan Dasbor Kinerja. Laporan Kinerja menyediakan identifikasi mudah dan akses ke hotspot IO, pernyataan SQL berbiaya tinggi, dan banyak lagi. Plus, dengan 1 klik, pengembang dapat melihat di mana mengoptimalkan kueri mereka dengan Rencana Penjelasan Visual yang ditingkatkan dan mudah digunakan.

MySQL Workbench kini menyediakan solusi yang lengkap dan mudah digunakan untuk memigrasikan Microsoft SQL Server, Microsoft Access, Sybase ASE, PostgreSQL, dan tabel, objek, dan data RDBMS lainnya ke MySQL. Pengembang dan DBA dapat dengan cepat dan mudah mengonversi aplikasi yang ada untuk dijalankan di MySQL baik di Windows maupun platform lainnya. Migrasi juga mendukung migrasi dari versi MySQL sebelumnya ke rilis terbaru. (Oracle 2023)

MySQL adalah Relational Database Management System (RDBMS) yang dapat diakses publik yang menggunakan bahasa Structured Query (SQL) untuk berinteraksi dengan database. MySQL menyimpan data dalam bentuk tabel yang dapat dimodifikasi menggunakan Structured Query Language. Adaptasinya dengan sistem komputasi yang berbeda seperti Windows, Linux, macOS, dan Ubuntu menjadikannya pilihan perangkat lunak RDBMS yang mudah digunakan.



Gambar 4. 1 Logo MySQL

Cara Install MySQL WorkBench

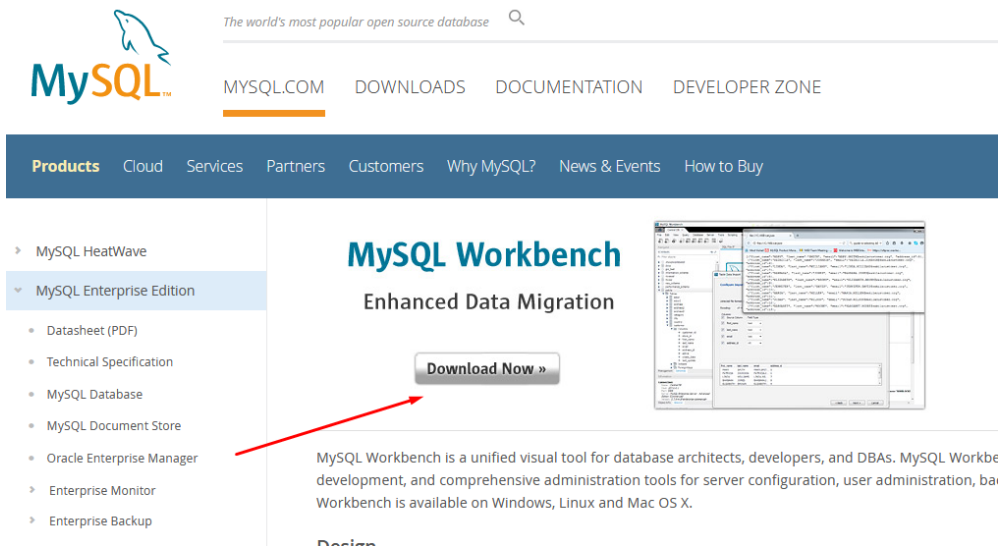
Selanjutnya, kita akan menginstal MySQL Workbench di Windows.

Proses instalasi mirip dengan sistem operasi lain.

1. Buka website MySQL di browser. Klik tautan berikut:

U<https://www.mysql.com/products/workbench/>

2. Pilih opsi Download.



Gambar 4. 2 Tampilan Download MySQL Work Bench

Pilih MySQL Installer for Windows.

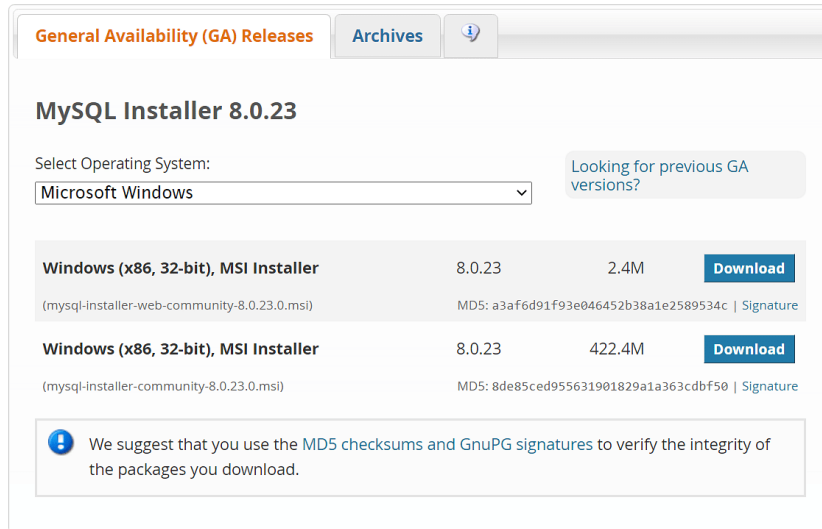
⊕ MySQL Community Downloads

- MySQL Yum Repository
- MySQL APT Repository
- MySQL SUSE Repository
- MySQL Community Server
- MySQL Cluster
- MySQL Router
- MySQL Shell
- MySQL Workbench
- MySQL Installer for Windows
- MySQL for Visual Studio
- C API (libmysqlclient)
- Connector/C++
- Connector/J
- Connector/NET
- Connector/Node.js
- Connector/ODBC
- Connector/Python
- MySQL Native Driver for PHP
- MySQL Benchmark Tool
- Time zone description tables
- Download Archives

ORACLE © 2021, Oracle Corporation and/or its affiliates

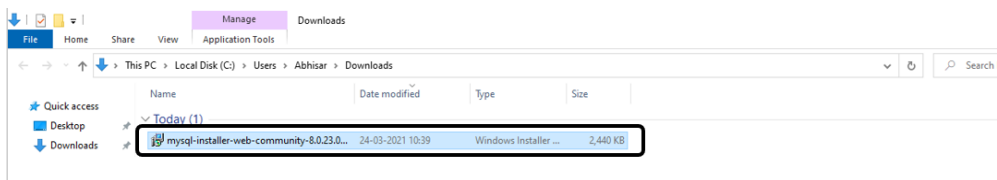
Gambar 4. 3 MySQL Community Download

Pilih the desired installer and click on download.



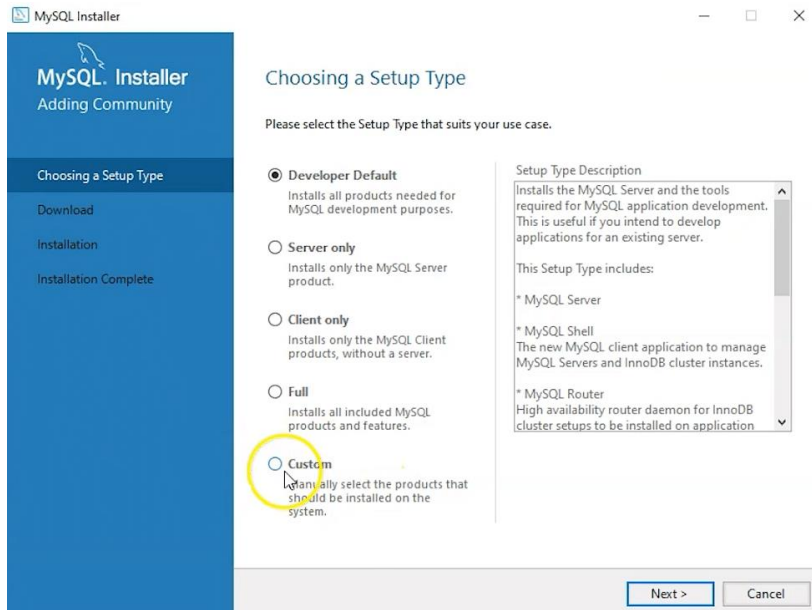
Gambar 4. 4 MySQL Installer

Setelah Selesai Download akan menampilkan file berikut dan klik file :



Gambar 4. 5 MySQL Installer Exe

Akan meminta izin; ketika itu terjadi, klik Ya. Pemasang kemudian akan terbuka. Sekarang, akan diminta untuk memilih jenis pengaturan. Di sini, pilih Custom.

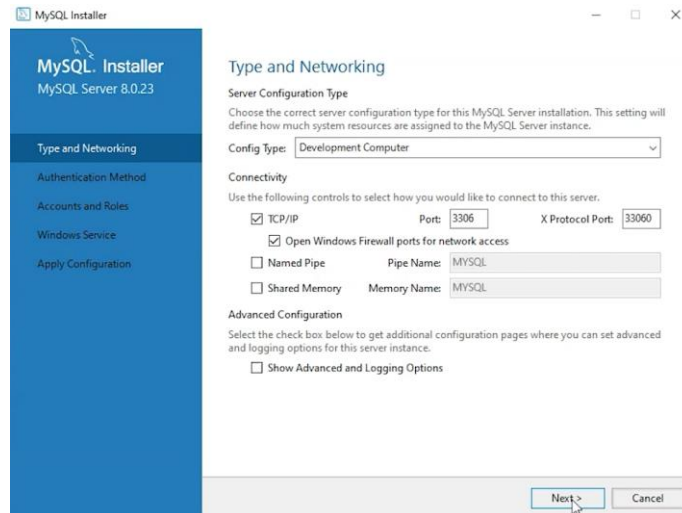


Gambar 4. 6 MySQL Installer

Klik Next. Dengan ini, Anda akan menginstal server MySQL, MySQL Workbench, dan MySQL shell.

Buka Server MySQL, pilih server yang ingin Anda instal, dan pindahkan ke bagian jendela Produk/Fitur yang akan diinstal. Sekarang, perluas Aplikasi, pilih MySQL Workbench dan MySQL shell. Pindahkan keduanya ke 'Products/Features to be installation'.

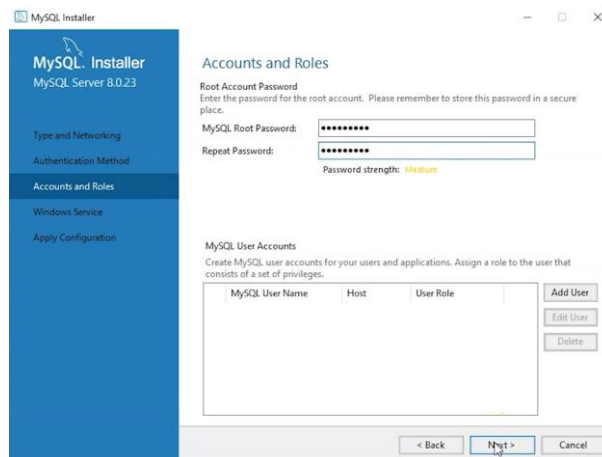
Klik tombol Berikutnya. Sekarang, klik tombol Execute untuk mengunduh dan menginstal server MySQL, MySQL Workbench, dan shell MySQL. (Simplilearn n.d.)



Gambar 4. 7 MySQL Installer

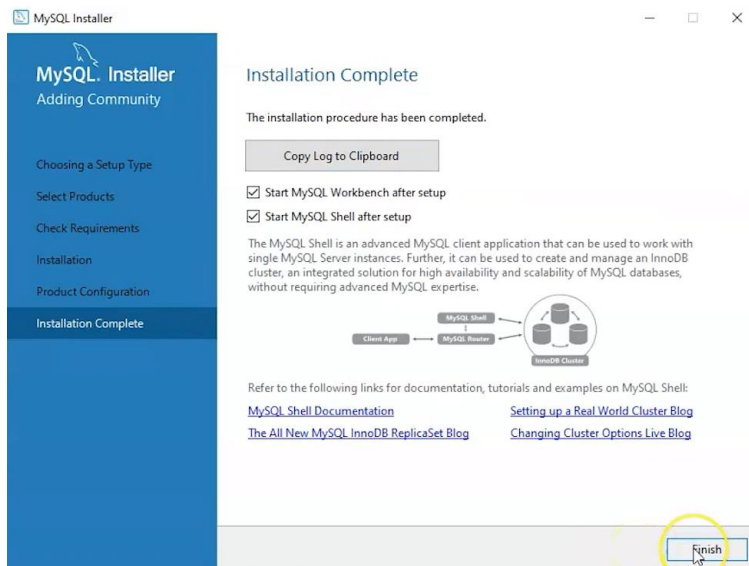
Setelah produk siap dikonfigurasi, klik Berikutnya. Di bawah Type and Networking, pilih pengaturan default dan pilih Next.

Untuk autentikasi, gunakan enkripsi kata sandi kuat yang direkomendasikan. Tetapkan kata sandi Root MySQL Anda dan klik berikutnya.



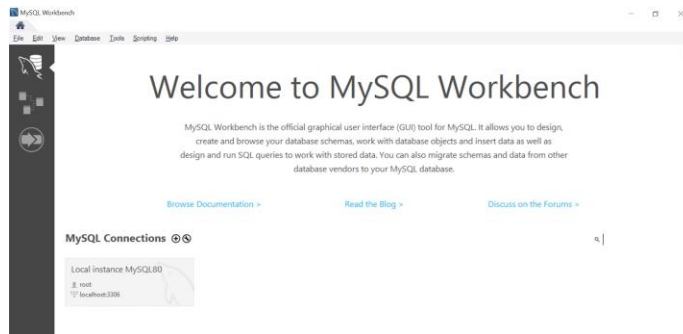
Gambar 4. 8 MySQL Installer

Buka pengaturan layanan windows default dan di bawah konfigurasi yang diterapkan, klik jalankan. Setelah konfigurasi selesai, klik selesai. Selesaikan penginstalan. Ini sekarang akan meluncurkan MySQL Workbench dan MySQL Shell.



Gambar 4. 9 Installation Complete

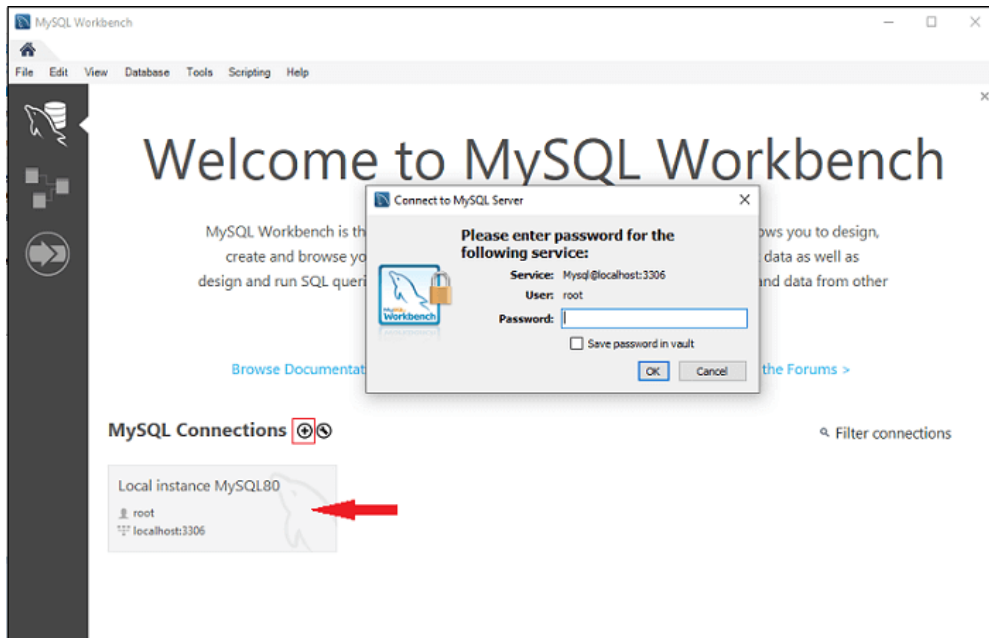
Setelah MySQL Workbench diinstal, pilih instance Lokal dan masukkan kata sandi. Sekarang, Anda dapat menggunakan tab kueri MySQL untuk menulis kueri SQL Anda.



Gambar 4. 10 Wellcome Message

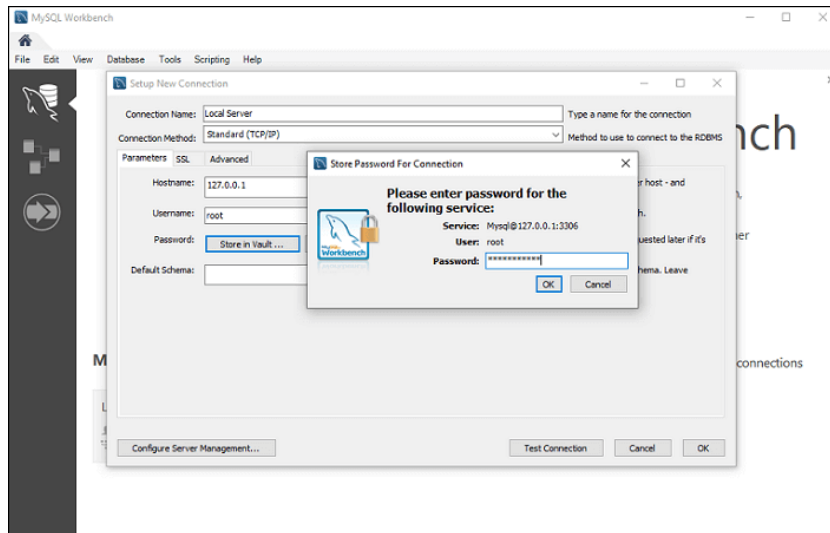
Cara Menggunakan MySQL WorkBench

Setelah Selesai Instalasi, maka akan menampilkan tampilan Welcome, MySQL Workbench akan terbuka di sistem Anda, seperti yang ditunjukkan pada layar di bawah ini. Di layar di atas, Anda perlu membuat sambungan. Untuk melakukan ini, klik dua kali kotak yang ditandai dengan panah merah. Di sini, Anda akan mendapatkan layar popup yang meminta untuk memasukkan kata sandi yang dibuat sebelumnya saat instalasi. Setelah memasukkan kata sandi, Anda dapat terhubung dengan Server.



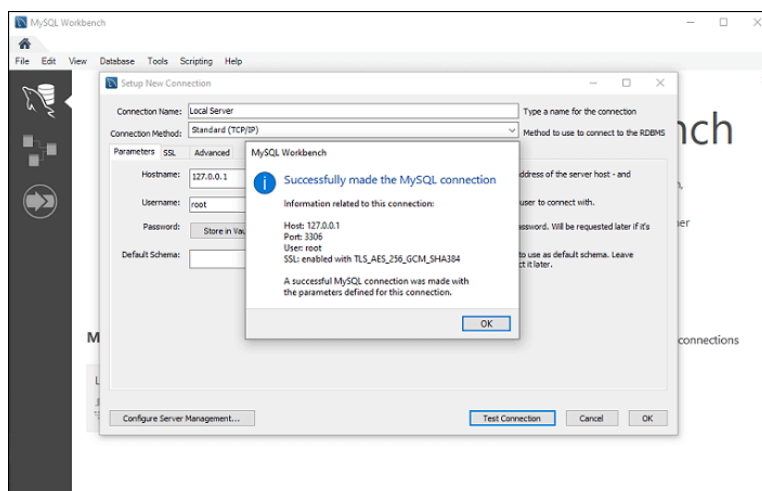
Gambar 4. 11 Input Password

Jika Anda tidak memiliki koneksi, Anda dapat membuat koneksi baru. Untuk membuat koneksi, klik ikon tambah (+) atau pergi ke bilah menu -> Database -> Hubungkan ke Database, layar berikut akan muncul. Sekarang, Anda harus mengisi semua detailnya. Di sini, Anda harus memastikan bahwa kata sandi yang dimasukkan harus sama dengan yang Anda buat sebelumnya.



Gambar 4. 12 Input User dan Password

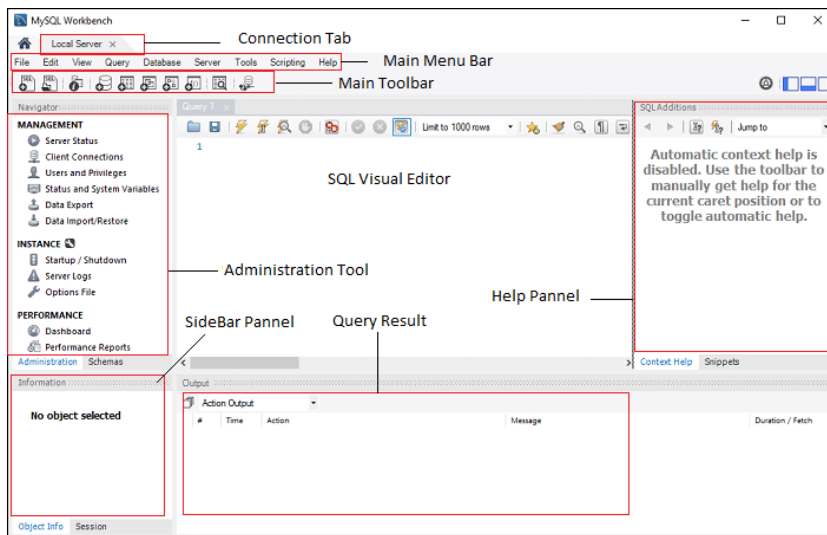
Setelah memasukkan semua detail, klik Test Connection untuk menguji konektivitas database. Jika koneksi berhasil, Anda akan mendapatkan layar berikut. Sekarang, klik tombol OK->OK untuk menyelesaikan pengaturan.



Gambar 4. 13 Success Login

Setelah Anda menyelesaikan semua pengaturan, itu akan membuka layar MySQL Workbench. Sekarang, klik dua kali pada koneksi yang

baru dibuat, Anda akan mendapatkan layar berikut di mana perintah SQL dapat dijalankan.



Gambar 4. 14 Dashboard Workbench

MySQL Workbench Administration Tool

The Administration Tool memainkan peran penting dalam mengamankan data perusahaan. Di sini, kita akan membahas manajemen pengguna, konfigurasi Server, cadangan dan pemulihan Database, log Server, dan banyak lagi.

Administrasi Pengguna adalah utilitas visual yang memungkinkan untuk mengelola pengguna yang terkait dengan instance MySQL Server yang aktif. Di sini, Anda dapat menambah dan mengelola akun pengguna, memberikan dan melepaskan hak istimewa, melihat profil pengguna, dan kedaluwarsa kata sandi. Konfigurasi Server Hal ini memungkinkan untuk konfigurasi lanjutan

dari Server. Ini memberikan informasi terperinci tentang Server dan variabel status, sejumlah utas, ukuran alokasi buffer, penyempurnaan untuk kinerja optimal, dan banyak lagi. Pencadangan dan pemulihan basis data Ini adalah alat visual, yang digunakan untuk mengimpor/mengekspor file dump MySQL. File dump berisi skrip SQL untuk membuat database, tabel, tampilan, dan prosedur tersimpan. Log Server Ini menampilkan informasi log untuk Server MySQL dengan setiap tab koneksi. Untuk setiap tab koneksi, ini menyertakan tab tambahan untuk log kesalahan umum. Dasbor Kinerja Tab ini memberikan tampilan statistik kinerja Server. Anda dapat membukanya dengan membuka tab Navigasi, dan di bawah bagian Performa, pilih Dasbor.

BAB V

Data Defenition Language & Data Manipulation Language

DATA DEFENITION LANGUAGE

Data Defenition Language (DDL) merupakan cara yang digunakan untuk melakukan Pengelolaan Struktur dari Database, baik struktur database secara utuh, maupun struktur tabel di dalam database.

Didalam DDL juga dikenal istilah Create, Read, Updata dan Delete.

Berikut adalah perintah MySQL yang digunakan untuk melakukan Create, Read, Update, dan Delete.

Create

Create digunakan untuk membuat Database, Membuat tabel, membuat kolom / field yang ada didalam tabel. Berikut adalah contoh penggunaan Create pada MySql :

```
CREATE DATABASE databasename;
```

Pada penggunaan Create database, *databasename* diganti dengan nama database yang akan dibuat.

```
CREATE TABLE Persons (  
    PersonID int,  
    LastName varchar(255),  
    FirstName varchar(255),  
    Address varchar(255),  
    City varchar(255)  
);
```

Query diatas digunakan untuk membuat tabel dengan 5 kolom yaitu kolom PersonId dengan tipe input Int, LastName dengan tipe input varchar, FirstName dengan tipe input varchar, Adress dengan tipe input varchar, City dengan tipe input varchar. Adapun untuk angka yang berwarna merah adalah maksimal input yang dapat di terima oleh kolom.

Read

Pada DDL, fungsi read menggunakan keyword SHOW dan digunakan untuk menampilkan database yang ada pada server lokal, tabel dan struktur tabel. Berikut adalah contoh penggunaan read pada DDL.

Untuk menampilkan database :

Show Databases ;

Untuk menampilkan tabel dan struktur tabel :

```
SHOW TABLES;
```

```
mysql> SHOW COLUMNS FROM City;
```

Field	Type	Null	Key	Default	Extra
ID	int(11)	NO	PRI	NULL	auto_increment
Name	char(35)	NO			
CountryCode	char(3)	NO	MUL		
District	char(20)	NO			
Population	int(11)	NO		0	

Update

Pada DDL, untuk melakukan Update data, dapat menggunakan perintah Alter, dimana pengguna dapat mengubah isi database, baik nama tabel, nama kolom, tipe input, dll. Berikut adalah penjelasan alter yang penulis kutip data W3School :

The ALTER TABLE statement is used to add, delete, or modify columns in an existing table.

The ALTER TABLE statement is also used to add and drop various constraints on an existing table.

Berikut adalah perintah Alter untuk DDL yang terdiri dari add kolom, edit kolom, hapus kolom:

```
ALTER TABLE Customers
ADD Email varchar(255);
```

Perintah diatas digunakan untuk menambah kolom email ke tabel customers.

```
ALTER TABLE Customers  
DROP COLUMN Email;
```

Perintah diatas digunakan untuk menghapus kolom Email dari tabel Customer.

```
ALTER TABLE table_name  
MODIFY COLUMN column_name datatype;
```

Perintah diatas merupakan perintah yang digunakan untuk memodifikasi kolom didalam tabel. Dimana untuk menggunakan perintah sql diatas maka Anda dapat mengganti *table_name* dengan nama tabel., *column_name* dengan nama kolom yang akan di modifikasi, dan *datatype* adalah tipe input.

```
ALTER TABLE Persons  
MODIFY COLUMN DateOfBirth year;
```

Berikut adalah kumpulan perintah mengganti kolom tabel pada mysql untuk alter pada DDL :

```
ALTER TABLE table_name RENAME COLUMN old_column_name TO new_column_name;
```



```
ALTER TABLE employees RENAME COLUMN id TO employ_id;
```

```
ALTER TABLE employees CHANGE id employ_id VARCHAR(25);
```

DELETE

Untuk menggunakan fungsi delete pada DDL, maka kita dapat menggunakan perintah DROP. DROP dapat digunakan untuk menghapus tabel, dan database.

```
DROP TABLE Shippers;
```

Catatan: Hati-hati sebelum menghapus tabel. Menghapus tabel akan mengakibatkan hilangnya informasi lengkap yang tersimpan dalam tabel!

DATA DATA MANIPULATION LANGUAGE

DATA MANIPULATION LANGUAGE Merupakan Query SQL yang secara eksplisit digunakan untuk melakukan perubahan pada DATA yang ada didalam database, seperti: operasi CRUD untuk membuat, membaca, memperbarui, dan menghapus data. Berbeda dengan DDL, DML tidak melakukan pengolahan Struktur dari database. Tapi fokus hanya untuk mengelola ISI data Database.

Menggunakan perintah INSERT, SELECT, UPDATE, dan DELETE.

Create

Create digunakan untuk mengisi data kedalam tabel, memasukan data ke tabel harus menyesuaikan jumlah data yang diinput sesuai dengan kolom. Dan pada DML, create menggunakan perintah INSERT Berikut adalah contoh penggunaan INSERT pada MySql :

```
INSERT INTO table_name (column1, column2, column3, ...)
VALUES (value1, value2, value3, ...);
```

Pada penggunaan Insert data, *table_name* diganti dengan nama table yang akan diinput, dan *column1*, 2, 3 dst diganti dengan nama kolom, dan *value* diganti dengan isi data yang akan diinput.

```
INSERT INTO Customers (CustomerName, ContactName, Address, City, PostalCode,
Country)
VALUES ('Cardinal', 'Tom B. Erichsen', 'Skagen 21', 'Stavanger', '4006',
'Norway');
```

Query diatas digunakan untuk mengisi data ke tabel Customers, dimana terdapat kolom CustomerName, ContactName, Address, City, PostalCode, Country. Dan data yang diisi adalah sebagai berikut :

92	Cardinal	Tom B. Erichsen	Skagen 21	Stavanger	4006	Norway
----	----------	--------------------	-----------	-----------	------	--------

Tugas :

Buat tabel dengan kolom diatas dan akan menghasilkan data seperti berikut ini :

CustomerID	CustomerName	ContactName	Address	City	PostalCode	Country
89	White Clover Markets	Karl Jablonski	305 - 14th Ave. S. Suite 3B	Seattle	98128	USA
90	Wilman Kala	Matti Karttunen	Keskuskatu 45	Helsinki	21240	Finland
91	Wolski	Zbyszek	ul. Filtrowa 68	Walla	01-012	Poland
92	Cardinal	Tom B. Erichsen	Skagen 21	Stavanger	4006	Norway

Read

Pada DML, fungsi read menggunakan keyword `SELECT` dan digunakan untuk menampilkan data yang ada pada tabel di database. Berikut adalah contoh penggunaan read pada DML.

```
SELECT * FROM table_name;
```

```
SELECT CustomerName, City, Country FROM Customers;
```

Update

Pada DML, untuk melakukan Update data, dapat menggunakan perintah `UPDATE`, dimana pengguna dapat mengubah data didalam tabel, berikut adalah penggunaan update pada data :

```
UPDATE table_name  
SET column1 = value1, column2 = value2, ...  
WHERE condition;
```

```
UPDATE Customers  
SET ContactName = 'Alfred Schmidt', City = 'Frankfurt'  
WHERE CustomerID = 1;
```

Delete

Pada DML, untuk perintah hapus menggunakan DELETE. Berikut adalah contoh penggunaan delete :

Example

```
DELETE FROM Customers WHERE CustomerName='Alfreds Futterkiste';
```

Untuk menghapus data dengan CustomerName = Alfreds Futterkiste.

Example

```
DELETE FROM Customers;
```

Untuk menghapus seluruh isi dari tabel customer .

TUGAS : Buatlah 3 buah tabel didalam database masing-masing yang memiliki data minimal 50 untuk satu tabel data utama, 5 untuk tabel kategori , dan 5 untuk tabel user admin.

BAB VI

WHERE CLAUSE PADA SQL

Setelah memahami perbedaan antara DDL dan DML, Selanjutnya pada bab ini kita akan membahas masalah where clause yang digunakan pada SQL mengenai Where Clause, apa saja yang dapat di lakukan dengan Kondisi-Kondisi tertentu pada data di sebuah tabel.

Text Fields vs. Numeric Fields

Untuk melakukan filter data dengan kondisi tertentu dibedakan terhadap tipe input dikolom yang menjadi filter, jika kolom yang menjadi filter memiliki tipe input text maka menggunakan kutip diantara teks dan untuk tipe input numerik tanpa menggunakan tanda kutip.

Example

```
SELECT * FROM Customers  
WHERE Country = 'Mexico';
```

Query diatas digunakan untuk memfilter data text berdasarkan kolom country, dimana data yang di tampilkan dari tabel customer adalah data dengan Country = Mexico.

Example

```
SELECT * FROM Customers
WHERE CustomerID = 1;
```

Query diatas digunakan untuk memfilter data berdasarkan kolom CustomerID dengan tipe input Numeric, dimana data yang di tampilkan dari tabel customer adalah data dengan CustomerID = 1.

Tabel 4.1 Tabel Operator

Operator	Fungsi
=	Setara / Sama Dengan
>	Lebih besar dari
<	Kurang dari
>=	Lebih besar dari atau sama
<=	Kurang dari atau sama
<>	Tidak sama. Catatan: Dalam beberapa versi SQL, operator ini dapat ditulis sebagai !=
BETWEEN	Antara rentang tertentu
LIKE	Cari pola dengan ketentuan tertentu
IN	Untuk menentukan beberapa kemungkinan nilai untuk kolom didalam sebuah list tertentu

Berikut penggunaan operator-operator pada tabel diatas :

```
SELECT * FROM Products  
WHERE Price > 30;
```

```
SELECT * FROM Products  
WHERE Price <> 18;
```

```
SELECT * FROM Products  
WHERE Price BETWEEN 50 AND 60;
```



```
SELECT * FROM Customers  
WHERE City LIKE 's%';
```

Untuk Like, operator LIKE digunakan untuk memfilter data dengan char tertentu, sebagai contoh di query diatas adalah 's%' dimana maksud nya adalah mengambil data dengan huruf awal s, dimana % adalah kata, jika huruf akhir s maka menggunakan '%s'.

```
SELECT * FROM Customers  
WHERE City IN ('Paris','London');
```

Pada operator in, data di filter berdasarkan list yang ada di dalam kurung setelah IN, contoh diatas adalah memilih data dengan city Paris atau London.

TUGAS : Implementasikan operator-operator tersebut ke database masing-masing.

MySQL AND, OR and NOT Operators

Operator MySQL AND, OR dan NOT Klausa WHERE dapat digabungkan dengan operator AND, OR, dan NOT. Operator AND dan OR digunakan untuk memfilter record berdasarkan lebih dari satu kondisi:

1. Operator AND menampilkan catatan jika semua kondisi yang dipisahkan oleh AND adalah BENAR.
2. Operator ATAU menampilkan catatan jika salah satu kondisi yang dipisahkan oleh ATAU adalah BENAR.
3. Operator NOT menampilkan catatan jika kondisi TIDAK BENAR.

Example

```
SELECT * FROM Customers  
WHERE Country = 'Germany' AND City = 'Berlin';
```

Example

```
SELECT * FROM Customers  
WHERE City = 'Berlin' OR City = 'Stuttgart';
```

Try it Yourself »

Example

```
SELECT * FROM Customers  
WHERE NOT Country = 'Germany';
```

SORTING DATA

Pengurutan adalah proses mengatur data menjadi urutan yang bermakna sehingga Anda dapat menganalisisnya dengan lebih efektif. Misalnya, Anda mungkin ingin mengurutkan data penjualan berdasarkan bulan kalender sehingga Anda dapat membuat grafik kinerja penjualan. Anda dapat menggunakan Penemu untuk mengurutkan data sebagai berikut:

1. mengurutkan data teks ke dalam urutan abjad
2. mengurutkan data numerik ke dalam urutan numerik

Mengelompokkan data urutan ke berbagai tingkatan, misalnya, Anda dapat mengurutkan Kota dalam Bulan dalam Tahun.

Pengurutan data lembar kerja juga memudahkan untuk dianalisis. Misalnya, Anda mungkin ingin mengurutkan data penjualan dari penjualan yang paling menguntungkan hingga penjualan yang paling tidak menguntungkan untuk menunjukkan posisi relatif dari produk terlaris perusahaan Anda.

Untuk mengurutkan data, pada MySQL menggunakan Order By baik untuk ascending maupun descending. Secara default, jika Anda tidak menggunakan Desc di akhir query, maka perintah akan otomatis ke ascending.

Example

```
SELECT * FROM Customers  
ORDER BY Country;
```

Example

```
SELECT * FROM Customers  
ORDER BY Country DESC;
```

Pada query diatas adalah contoh untuk mengurutkan data berdasarkan kolom Country, baik dengan Ascending maupun Descending.

LIMIT Clause

LIMIT digunakan untuk menentukan jumlah record yang akan dikelola. Klausula LIMIT berguna pada tabel besar dengan ribuan record. Menampilkan data dalam jumlah besar dapat memengaruhi kinerja dari sistem, untuk itu perlu mengambil data dalam jumlah tertentu sesuai dengan kebutuhan.

Example

```
SELECT * FROM Customers  
LIMIT 3;
```

Example

```
SELECT * FROM Customers  
WHERE Country='Germany'  
LIMIT 3;
```

Untuk penggunaan Limit pada tabel yang diurutkan dapat menggunakan query berikut:

SQL Statement:

```
SELECT * FROM Customers  
Order By Country Desc LIMIT 3;
```

MIN() and MAX() Functions

Min merupakan perintah untuk memilih nilai terkecil dari tabel tertentu dan Max untuk memilih data terbesar dari tabel tertentu berdasarkan kolom.

Example

```
SELECT MIN(Price) AS SmallestPrice  
FROM Products;
```

Example

```
SELECT MAX(Price) AS LargestPrice  
FROM Products;
```

COUNT(), AVG() and SUM() Functions

Count bertugas untuk menghitung jumlah baris data dari kolom tertentu.

AVG bertugas untuk menghitung rata-rata dari total data yang di pilih dari kolom tertentu

SUM bertugas menjumlahkan nilai dari kolom dengan tipe input numerik.

Example

```
SELECT COUNT(ProductID)
FROM Products;
```

Example

```
SELECT AVG(Price)
FROM Products;
```

Example

```
SELECT SUM(Quantity)
FROM OrderDetails;
```

Pada tabel berikut jelaskan apa perbedaan dari Count dan SUM jika digunakan pada Kolom Quantity:

OrderDetailID	OrderID	ProductID	Quantity
1	10248	11	12
2	10248	42	10
3	10248	72	5
4	10249	14	9
5	10249	51	40

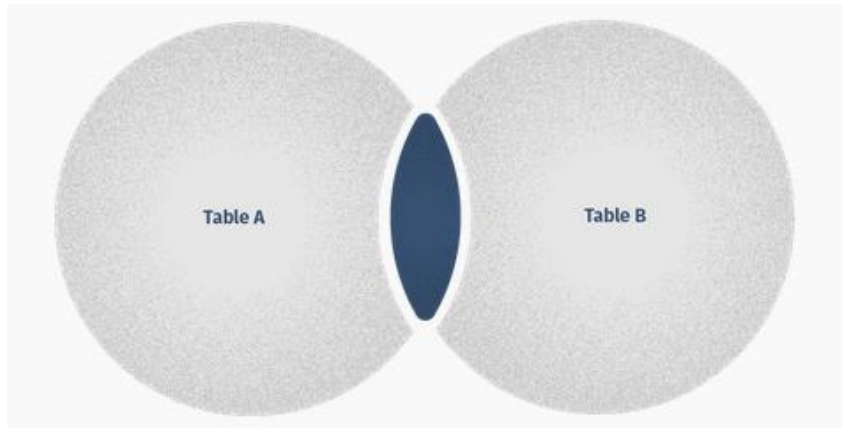
BAB VII

JOIN PADA TABEL

Dalam database relasional, seperti SQL Server, Oracle, MySQL, dan lainnya, data disimpan dalam beberapa tabel yang terkait satu sama lain dengan nilai kunci yang sama. Oleh karena itu, ada kebutuhan konstan untuk mengekstrak record dari dua atau lebih tabel ke dalam tabel hasil berdasarkan beberapa kondisi. Di SQL Server, ini dapat dengan mudah dilakukan dengan klausa SQL JOIN. Join adalah klausa SQL yang digunakan untuk meminta dan mengakses data dari beberapa tabel, berdasarkan hubungan logis antara tabel tersebut. Dengan kata lain, Join menunjukkan bagaimana SQL Server harus menggunakan data dari satu tabel untuk memilih baris dari tabel lain.

INNER JOIN

Inner Join merupakan metode penggabungan data dimana baris yang memiliki nilai yang cocok/sama saja yang digunakan untuk mengambil data yang muncul di kedua tabel.



Gambar 7. 1 Inner Join Ilustrasi

(Devart n.d.)

Contoh Tabel yang menggunakan Inner join :

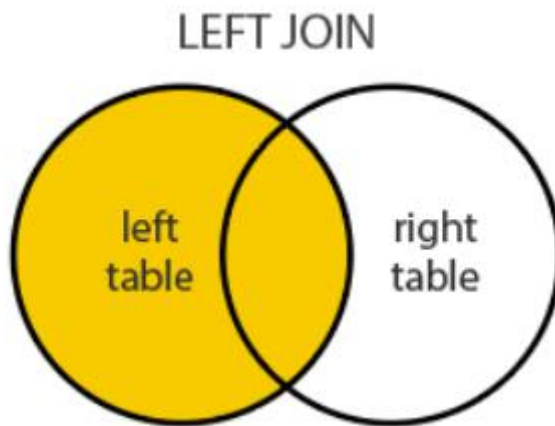
Table User			
id	name		
1	Salsa		
2	Gifa		
3	Rian		
4	Reyvan		
Table Order			
id	name	total	
1	2	50000	
2	2	75000	
3	3	15000	
4	4	8000	
5	10	99000	

Silahkan masukkan query berikut ini ke mysql workbench kalian untuk menampilkan hasil inner join :

```
SELECT user.name, order.total FROM order INNER JOIN
user ON user.id = order.user_id;
```

LEFT JOIN

Left Join adalah metode penggabungan yang digunakan untuk memilih semua baris dari tabel kiri dan juga nilai yang cocok di antara dua tabel dan jika data dari tabel kiri tidak memiliki kecocokan dari tabel kanan, maka isi dari kolom yang diambil dari sebelah kanan adalah *null*.



Gambar 7. 2 Ilustrasi Left Join

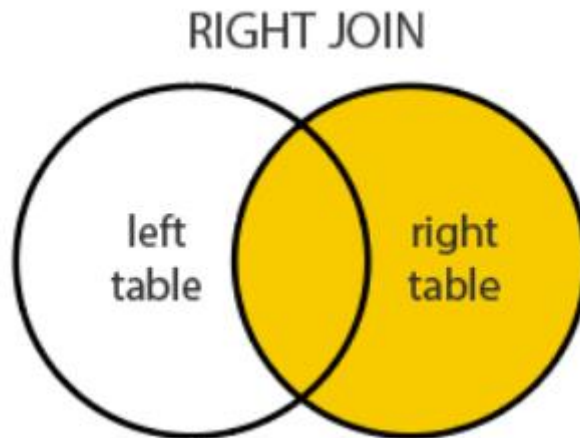
Masukkan perintah berikut ke *mysql workbench* anda dan lihat hasil yang akan ditampilkan :

```
SELECT user.name, order.total FROM order LEFT JOIN  
user ON user.id = order.user_id;
```

RIGHT JOIN

Right Join adalah metode penggabungan yang digunakan untuk memilih semua baris dari tabel kanan dan juga nilai yang cocok di antara dua tabel dan jika data dari tabel kanan tidak memiliki

kecocokan dari tabel kiri, maka isi dari kolom yang diambil dari sebelah kiri adalah *null*. (Point 2022)



Gambar 7. 3 Ilustrasi Right Join (DQLab 2022)

Masukkan perintah berikut ke *mysql workbench* anda dan lihat hasil yang akan ditampilkan :

```
SELECT user.name, order.total FROM order RIGHT JOIN  
user ON user.id = order.user_id;
```

BAB VIII

PROCEDURE PADA MYSQL

Procedure adalah kumpulan pernyataan SQL yang telah dikompilasi yang disimpan di dalam database. Ini adalah subrutin atau subprogram yang disimpan dan dapat digunakan untuk mempersingkat penulisan query. Prosedur selalu berisi nama, daftar parameter, dan pernyataan SQL. Kita dapat memanggil prosedur dengan menggunakan trigger, prosedur dan aplikasi lain seperti Java, Python, PHP, dll. Ini pertama kali diperkenalkan di MySQL versi 5.

Saat ini, dapat didukung oleh hampir semua sistem basis data relasional. Jika kita mempertimbangkan aplikasi perusahaan, kita selalu perlu melakukan tugas khusus seperti pembersihan basis data, memproses penggajian, dan banyak lagi pada basis data secara teratur. Tugas tersebut melibatkan beberapa pernyataan SQL untuk menjalankan setiap tugas. Proses ini mungkin mudah jika kita mengelompokkan tugas-tugas ini menjadi satu tugas. Kami dapat memenuhi persyaratan ini di MySQL dengan membuat prosedur tersimpan di database kami.

Fitur Store Procedure

Stored Procedure meningkatkan kinerja aplikasi. Setelah prosedur tersimpan dibuat, mereka dikompilasi dan disimpan dalam database.

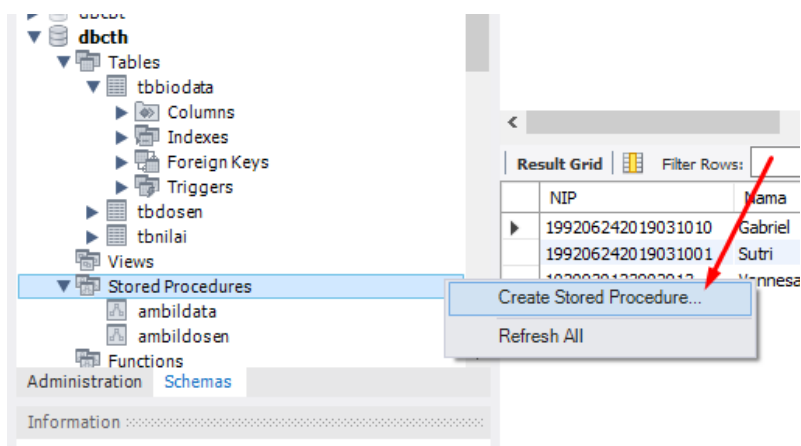
Procedur tersimpan mengurangi lalu lintas antara aplikasi dan server basis data. Karena aplikasi harus mengirim hanya nama dan parameter prosedur tersimpan alih-alih mengirim beberapa pernyataan SQL. (SoftwareTestingHelp 2023) MySQL Menyediakan Stored Procedure untuk memiliki kumpulan pernyataan MySQL yang

dikelompokkan bersama dalam sebuah fungsi yang dapat dipanggil sesuai permintaan dengan parameter masukan tertentu.

Dengan procedure, Anda dapat menggunakan kembali kode dan skrip MySQL yang menghemat banyak waktu dan upaya serta memberikan kualitas dan kinerja tinggi dengan struktur termodulasi ke Skrip MySQL.

Prosedur tersimpan dapat digunakan kembali dengan mudah untuk aplikasi apa pun. Sebuah procedure selalu aman. Administrator basis data dapat memberikan izin untuk aplikasi yang mengakses prosedur tersimpan dalam basis data tanpa memberikan izin apa pun pada tabel basis data.

Bagaimana cara membuat prosedur?



Gambar 8. 1 Menu untuk masuk ke Store Procedur

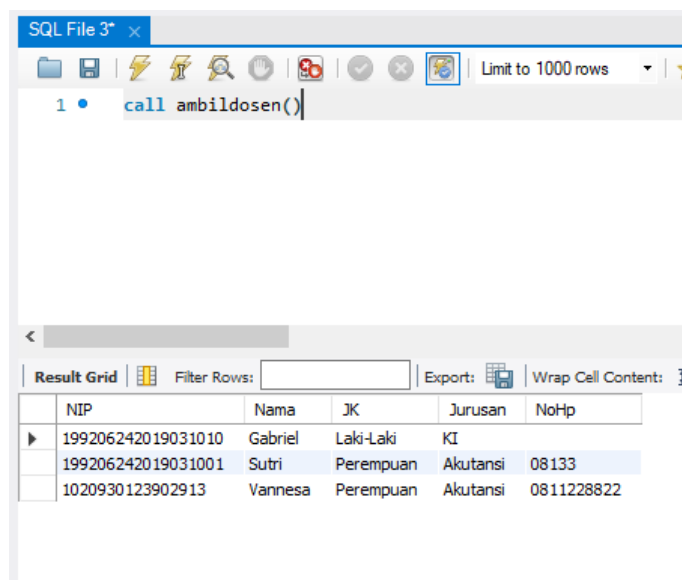
Masukkan Sintaks berikut digunakan untuk membuat procedure tersimpan di MySQL.

```

CREATE DEFINER=`root`@`localhost` PROCEDURE `ambildosen`()
BEGIN
select * from tbdosen order by nama ASC;
END

```

Setelah membuat procedure, maka dapat memanggil dengan cara berikut :



The screenshot shows a SQL IDE window titled "SQL File 3". The editor contains the command `call ambildosen();`. Below the editor, the "Result Grid" displays the output of the procedure call. The results are organized into a table with six columns: NIP, Nama, JK, Jurusan, and NoHp. Three rows of data are visible.

NIP	Nama	JK	Jurusan	NoHp
199206242019031010	Gabriel	Laki-Laki	KI	
199206242019031001	Sutri	Perempuan	Akutansi	08133
1020930123902913	Vannesa	Perempuan	Akutansi	0811228822

BAB IX

SERANGAN PADA MYSQL

Jumlah kejahatan dunia maya yang dilakukan terhadap bisnis dan aplikasi online telah meningkat dalam beberapa tahun terakhir. Penyerang dapat memanfaatkan bisnis dengan sistem yang dibangun dengan buruk untuk menimbulkan kerugian data dan moneter.

MySQL adalah salah satu sistem manajemen basis data relasional sumber terbuka paling populer di luar sana, dan digunakan untuk menyimpan data untuk situs web dan server web. Kerangka web dan server web paling populer menggunakan MySQL sebagai database default mereka.

MySQL memiliki beberapa kerentanan yang harus kami waspadai untuk mengamankan data kami dan dalam artikel ini, kami akan menjelajahi berbagai kerentanan ini dan menemukan perbaikannya.

SQL Injection

SQL Injection adalah salah satu serangan dunia maya paling berbahaya yang ditargetkan pada basis data untuk mencuri informasi sensitif atau menyebabkan hilangnya data. Ini adalah jenis serangan injeksi yang memungkinkan untuk menjalankan kueri SQL berbahaya yang dapat menyebabkan pencurian data atau kehilangan data.

SQL Injection adalah mimpi buruk bagi pengembang web karena mereka menjadi sasaran paling banyak karena celah yang diketahui tersedia di server dan aplikasi mereka. Pengguna jahat dapat menggunakan kerentanan SQL Injection untuk mem-bypass keamanan aplikasi dan prosedur otentikasi.

Penyerang yang berhasil dapat mengakses bagian aplikasi web atau server web yang diautentikasi dan diotorisasi, mengambil rekaman database, atau menambah, mengubah, dan menghapus rekaman di database.

Mari kita asumsikan kita memiliki sebuah aplikasi dan kita mencoba mengautentikasi seorang pengguna bernama John. Kami akan mengirimkan detail loginnya dalam kueri di bawah ke server untuk autentikasi:

```
Select * From Users Where Username='John' And Password='John123'
```

Aplikasi Anda pada dasarnya membuat kueri di atas dengan masukan pengguna untuk mengetahui apakah ada pengguna dengan nama pengguna dan kata sandi tersebut.

Sekarang anggaplah kita memiliki pengguna jahat yang memasukkan kata sandi jahat –*’ ATAU ‘1’=’1’. Saat aplikasi Anda membangun kueri basis data, itu akan menjadi:

```
Select * From Users Where Username='John' And Password='*' Or '1'='1'
```

Menjalankan kueri ini akan selalu mengembalikan true dan menganggap kata sandi pengguna benar. Bagian pertama akan mencoba mencari pengguna – john yang memiliki kata sandi – * dan itu pasti tidak akan menghasilkan apa-apa atau salah. Namun, bagian kedua akan selalu mengembalikan true, sehingga membuat kueri lolos.

Ini pada dasarnya akan melewati otentikasi apa pun karena kueri akan mengembalikan true dan setiap pengguna yang cocok dengan nama pengguna john.

Sekarang, mari kita lihat skenario lain di mana pengguna jahat dapat mengambil tabel lain di database.

```
Select * From Users Where Username='John' Union (Select 1, Table_Name, Table_Schema From Information_Schema.Tables);
```

Dengan kueri di atas, pengguna dapat mengambil semua tabel dari database. Sekarang setelah penyerang mengetahui semua tabel tempat Anda menyimpan data, mereka dapat menanyakan kolom dari setiap tabel serta setiap data di setiap tabel.

Penyerang akan menggunakan kueri di bawah ini untuk menanyakan kolom pengguna:

```
Select * From Users Where Username='John' Union (Select 2, Column_Name, 3 From Information_Schema.Columns Where Table_Name = 'Users');
```

Kemudian ini untuk menanyakan semua data di tabel pengguna:

```
Select * From Users Where Username='John' Union (Select Id, Username, Password From Users);
```

Penyerang dapat menjadikan john sebagai admin di database bahkan tanpa menggunakan kata sandi yang tepat untuk akun seperti di bawah ini:

```
SELECT id FROM users WHERE username = 'john';
```

```
INSERT INTO group_membership (id, group) VALUES (SELECT id FROM users WHERE username='john', 'Administrator')
```

Semua ini dapat dihindari dan diperbaiki dengan membersihkan setiap input pengguna yang masuk dan juga menggunakan pernyataan yang disiapkan SQL untuk membuat kueri berparameter.

Cara terbaik untuk mencegah injeksi SQL adalah dengan tidak memercayai input pengguna dan menggunakan pernyataan yang disiapkan karena input sedang dibersihkan sebelum menghasilkan kueri akhir.

Cross Site Scripting (XSS) – Client Hijacking

Cross-Site Scripting (XSS) adalah salah satu serangan dunia maya paling umum dan berbahaya yang menargetkan aplikasi web. Ini biasanya dilakukan di sisi klien aplikasi (biasanya browser) dengan menggunakan skrip jahat yang disematkan pada halaman web di aplikasi web untuk mengambil kendali halaman saat ini.

Ini mengeksploitasi kerentanan situs web dengan menyuntikkan skrip berbahaya yang akan berjalan di sisi klien. Idennya adalah untuk menyematkan dan menjalankan skrip di halaman web aplikasi web dan menjalankannya dalam pola yang diinginkan oleh penyerang.

Skrip dapat dijalankan setiap kali halaman dimuat, atau setiap kali acara terkait dilakukan.

Ini sangat mirip dengan injeksi SQL. Perbedaannya di sini adalah kami menggunakan kode sisi klien yang sebenarnya (javascript, html, css) pada XSS dan kueri SQL pada injeksi SQL. Dalam hal ini, pengguna jahat dapat mengelabui database untuk menyimpan skrip ini sebagai string jika aplikasi menampilkan konten di halaman web.

Anggaplah Anda memiliki aplikasi yang menerima komentar pengguna dan menampilkannya. Anda perlu mengambil komentar pengguna sebagai string dan menyimpannya di database. Kemudian di sisi klien, Anda dapat menampilkan ke bagian komentar.

Namun, jika pengguna jahat memasukkan komentar yang memiliki skrip, jika komentar tersebut ditampilkan di halaman web tanpa membersihkan dan keluar dari komentar saat menyimpannya, skrip akan terlihat sebagai elemen html yang akan dieksekusi.

Anggaplah Anda memasukkan ini di kotak komentar:

```
<script>alert('Ini adalah XSS Injection')</script>
```

Dan katakanlah kita ingin membuat daftar komentar seperti di bawah ini:

```
<div class="comments">
  <div class="comment">
    <h5>Author: Toby</h5>
    <p>Comment: New comment</p>
  </div>
  <div class="comment">
    <h5>Author: Emma</h5>
    <p>Comment: New comment</p>
  </div>
  <div class="comment">
    <h5>Author: Anonymous</h5>
    <p>Comment: <script>alert('This is XSS Injection')
  </script></p>
  </div>
</div>
```

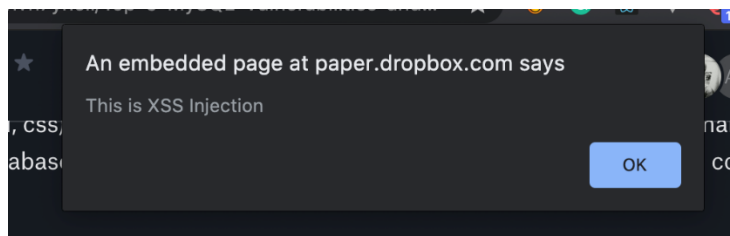
Gambar 9. 1 List Comment

Jika komentar disimpan ke dalam database tanpa keluar darinya dan kami mencoba merendernya di halaman web, itu akan menampilkan komentar seperti itu dan menjalankan skrip apa pun:

Itu berarti Anda benar-benar dapat mengontrol halaman saat ini dengan javascript serta mengubah seluruh konten halaman web atau mengarahkan pengguna ke situs web jahat atau halaman phishing.

Untuk memperbaikinya, Anda harus keluar dari semua input sebelum menyimpan dan sebelum merender di halaman web atau menghapus semua konten html atau mencegah pengguna memasukkan konten html apa pun. Bergantung pada tumpukan dan teknologi aplikasi web Anda, ada banyak cara untuk keluar dari konten html apa pun.

Jika Anda keluar dari skrip di atas, Anda akan mendapatkan sesuatu seperti ini di database Anda:



Gambar 9. 2 Fishing Alert

Itu berarti Anda benar-benar dapat mengontrol halaman saat ini dengan javascript serta mengubah seluruh konten halaman web atau mengarahkan pengguna ke situs web jahat atau halaman phishing. Untuk memperbaikinya, Anda harus keluar dari semua input sebelum menyimpan dan sebelum merender di halaman web atau menghapus semua konten html atau mencegah pengguna memasukkan konten html apa pun. Bergantung pada tumpukan dan teknologi aplikasi web Anda, ada banyak cara untuk keluar dari konten html apa pun.

CVE-2012-5614 – Denial of Service

Serangan penolakan layanan adalah salah satu serangan dunia maya paling populer dan berbahaya yang ditargetkan pada server web atau contoh server seperti MySQL. Tujuannya adalah untuk membuat layanan, sumber daya jaringan, atau instans tidak tersedia untuk sementara atau tanpa batas, mengganggu keseluruhan sistem atau host yang terhubung dengan sumber daya atau instans.

Dalam kasus MySQL, ini dapat menyebabkan instance MySQL macet sehingga membuatnya tidak dapat digunakan untuk sementara oleh layanan apa pun yang terhubung dengannya untuk sumber data.

Kerentanan ini memungkinkan pengguna yang diautentikasi dari jarak jauh untuk menyebabkan penolakan layanan (kerusakan MySQL) melalui perintah SELECT yang berbahaya dengan perintah UpdateXML yang berisi XML yang disengaja dengan sejumlah besar elemen bersarang yang unik.

Hal ini membuat MySQL rentan terhadap kerentanan denial-of-service. Pengguna jahat dapat mengeksploitasi kerentanan ini untuk merusak database, menolak akses ke pengguna yang tepat.

Menjalankan kueri SQL di bawah ini akan merusak basis data Anda:

```
SELECT      UpdateXML('<a>$a<b>ccc</b><d></d></a>',      '/a',  
'<e>fff</e>') AS val1
```

Untuk memperbaiki kerentanan ini, pastikan Anda menggunakan versi MySQL yang lebih besar dari 5.5.* karena 5.1.67 dan sebelumnya, 5.5.29 dan sebelumnya dipengaruhi oleh kerentanan.

CVE-2016-6663 – Race Condition

Kondisi balapan adalah situasi atau skenario yang tidak diinginkan yang terjadi ketika perangkat, sistem, atau perangkat lunak mencoba menjalankan dua atau lebih operasi secara bersamaan, namun karena

sifat sistem atau perangkat, operasi harus dijalankan dalam urutan atau waktu yang tepat karena ke peristiwa tak terkendali lainnya untuk memastikannya dilakukan secara efisien.

Dalam kasus MySQL, kerentanan ini dapat menyebabkan kondisi balapan yang bisa sangat parah. Itu dapat membiarkan pengguna lokal yang memiliki akses ke database meningkatkan hak istimewa mereka dan menjalankan kode arbitrer sebagai pengguna lokal database.

Race condition hadir di Oracle MySQL sebelum 5.5.52, 5.6.x sebelum 5.6.33, 5.7.x sebelum 5.7.15, dan 8.x sebelum 8.0.1; MariaDB sebelum 5.5.52, 10.0.x sebelum 10.0.28, dan 10.1.x sebelum 10.1.18 dan memungkinkan pengguna lokal dengan izin tertentu untuk mendapatkan hak istimewa dengan memanfaatkan penggunaan `my_copystat` dengan `REPAIR TABLE` untuk memperbaiki tabel MyISAM.

Pengguna jahat dapat mengeksploitasi kerentanan ini untuk melewati batasan keamanan tertentu dan menjalankan tindakan tidak sah. Ini dapat membantu meluncurkan serangan berbahaya lebih lanjut.

Masalah telah diperbaiki dan ditambal di versi yang terpengaruh yang dinyatakan sebelumnya dan versi baru berikutnya, Oracle MySQL 5.5.x, 5.6.x, 5.7.x, dan seterusnya.

Anda dapat membaca lebih lanjut di sini.

CVE-2012-5615 — Remote Preauth User Enumeration

Salah satu kerentanan yang sangat unik yang hanya terjadi ketika input yang dihasilkan pengguna tidak dibersihkan dengan baik sebelum diteruskan ke kueri database. Ini adalah jenis serangan yang ditargetkan pada kerentanan Enumerasi Pengguna jarak jauh di database server MySQL.

Tujuannya di sini adalah agar penyerang memvalidasi jika pengguna atau nama pengguna tertentu valid atau ada di basis data Anda dan menemukannya untuk mengakses informasi sensitifnya dengan menggunakan pesan kesalahan yang dihasilkan. Pengguna jahat dapat memanfaatkan kerentanan ini untuk menghitung nama pengguna yang valid dan mendapatkan akses tidak sah ke data sensitif.

Jadi dengan kerentanan ini, pengguna jahat dapat mengonfirmasi jika nama pengguna tertentu sedang digunakan oleh instance SQL db. Itu merespons dengan pesan kesalahan "Akses ditolak" jika akun tidak ada. Namun jika mengembalikan respons lain, penyerang kemudian dapat mengonfirmasi bahwa pengguna benar-benar ada.

Kerentanan ini memengaruhi versi berikut: Oracle MySQL 5.5.38 dan sebelumnya, 5.6.19 dan sebelumnya.

Untuk menghindarinya, Anda perlu membersihkan input yang dibuat pengguna dan memutakhirkan ke versi yang lebih baru setelah yang disebutkan. (Coding 2022)

BAB X

PENCADANGAN DATA PADA MYSQL

Salah Satu tugas dari database administrator adalah melakukan pencadangan database atau backup. Pencadangan dapat dilakukan secara rutin untuk mencegah kehilangan dan kerusakan data, bahkan siklus pencadangan otomatis saat ini sudah banyak diterapkan di berbagai sistem.

Jenis Backup di MySQL

Di MySQL, ada berbagai jenis cadangan. Strategi pencadangan mana yang akan dipilih bergantung pada beberapa faktor, misalnya, ukuran data, perangkat keras yang Anda gunakan, kinerja yang ingin Anda capai, kapasitas penyimpanan database Anda, dll. Selain itu, Anda harus mempertimbangkan berapa lama waktu yang diperlukan untuk mengembalikan cadangan. MySQL mendukung jenis pencadangan berikut:

Logical Backup

menampilkan struktur database dalam file .sql dengan membuat pernyataan CREATE atau INSERT. Nantinya, file ini bisa dikembalikan dengan bantuan utilitas mysqldump. Jenis ini hanya mencadangkan data tanpa indeks, sehingga memiliki ukuran yang kecil. Namun, opsi pemulihannya lebih lambat dibandingkan alternatifnya, karena harus menjalankan semua pernyataan satu per satu.

Physical Backup

menyalin file database dalam format yang sama, yang disimpan di disk. Ini lebih cepat daripada tipe logis tetapi hanya dapat dikembalikan ke server MySQL dari mesin database yang sama.

Consistent

Merupakan Cadangan yang konsisten menyalin file pada saat yang tepat – hanya setelah server berhenti berjalan atau terkunci.

Jika mungkin terjadi gangguan selama operasi penyalinan, pencadangan dapat dibagi menjadi beberapa jenis berikut:

1. Cold Backup memblokir akses ke data selama pencadangan dan tidak mengizinkan perubahan apa pun pada data. Ini sederhana, cepat, dan tidak memengaruhi kinerja.
2. Hot BackUp menyalin file saat database sedang berjalan. Pengguna dapat membaca dan memanipulasi data.
3. Warm BackUp membuat salinan basis data saat sedang berjalan. Pengguna dapat membaca tetapi tidak dapat menulis data selama pencadangan.

Apakah data harus disalin seluruhnya atau sebagian, pencadangan dapat diklasifikasikan ke dalam:

1. Full backup menyalin semua data dari database. Itu bisa logis atau fisik. Cadangan lengkap dapat dipulihkan di server lain.

2. Differential Backup menyalin semua perubahan yang dibuat sejak cadangan lengkap terbaru. Cadangan diferensial hanya dapat dipulihkan setelah cadangan lengkap dipulihkan.
3. Incremental menyalin semua perubahan data yang dibuat sejak cadangan sebelumnya (baik penuh atau diferensial).

Perintah mysqldump

Untuk mencadangkan database MySQL, Anda dapat menggunakan alat pihak ketiga atau menjalankan perintah mysqldump dari baris perintah.

mysqldump adalah utilitas baris perintah yang digunakan untuk menghasilkan cadangan basis data logis MySQL. Itu membuat file .sql tunggal yang berisi sekumpulan pernyataan SQL. File ini dapat digunakan untuk membuat tabel, objek, dan/atau menyisipkan data yang disalin dari database. Dengan utilitas, Anda dapat membuang tabel, satu atau beberapa database, atau seluruh server dengan semua database dan tabel beserta objeknya atau memigrasikannya ke server lain.

Selain itu, utilitas mysqldump dapat menampilkan dump ke format file CSV atau XML. Perlu dicatat bahwa mysqldump tidak dapat membuang database atau data ke file .sql yang terpisah.

Untuk memulihkan database, utilitas menjalankan semua pernyataan SQL untuk membuat ulang tabel dan mengisinya dengan data yang, pada gilirannya, membutuhkan banyak waktu.

Sintaks dasar perintah mysqldump mencakup parameter berikut:

```
mysqldump -u [user name] -p [password] -h [host name] [options] [database_name]  
[tablename] > [dumpfilename.sql]
```

Contoh menunjukkan bagaimana menggunakan mysqldump untuk membuat cadangan semua database MySQL di server. Skrip database diekspor ke file .sql, yang nantinya dapat Anda gunakan, misalnya, untuk memigrasikan database ke server baru. Untuk informasi selengkapnya, lihat Mencadangkan semua database server MySQL atau hanya yang diperlukan.

```
mysqldump --host=dbfmylast --user=root --port=3306 --p --all-databases >  
D:\backup_all_databases.sql
```

MySQL Workbench to back up a database

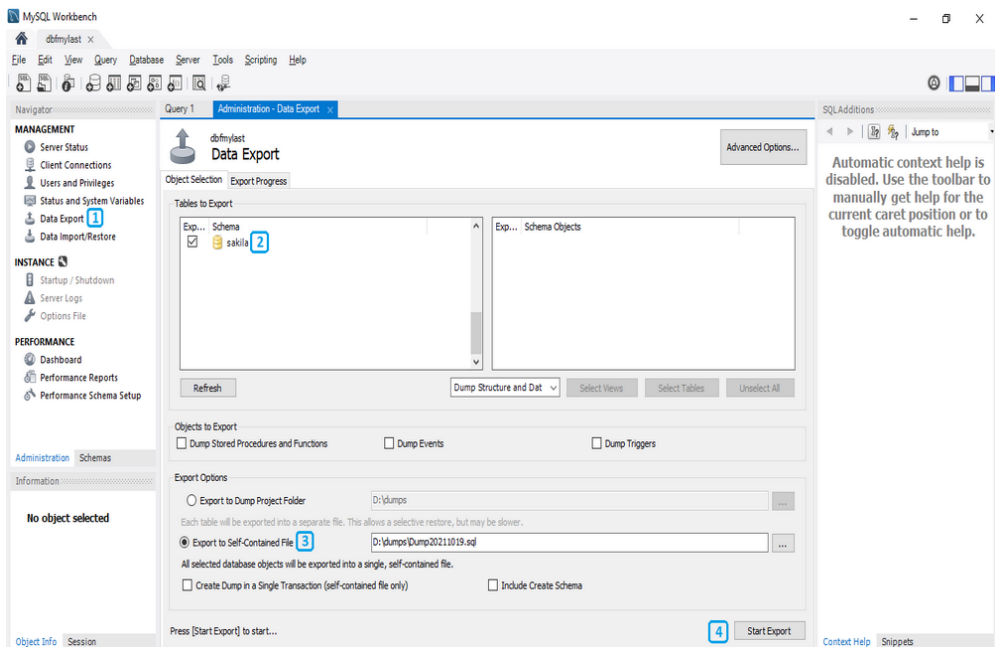
Pada bagian ini, kita akan mempelajari cara membuat cadangan basis data menggunakan alat MySQL Workbench.

Untuk membuat cadangan database, lakukan hal berikut:

1. Pada Administration panel, click Data Export. The Administration – Data Export terbuka.
2. Pada Object Select > Tables to Export tab, Pilih sakila schema.

3. Dibawah Export Options, pilih Export to Dump Project Folder jika Anda ingin tabel database disimpan ke file .sql terpisah atau Export to Self-Contained File untuk menyimpan dump database dalam satu file .sql.

4. Untuk membuat file cadangan, klik Start Export. Di bilah progres, Anda dapat melihat bahwa data telah diekspor ke file yang ditentukan.



Gambar 10. 1 Backup Database

TUGAS PBL

Implementasikan bab VI , Bab VII, Bab VIII dengan data berikut ini :



Pilih 1 sheet data untuk masing-masing kelompok, tulis hasil pekerjaan ke dalam log book dengan kolom yang terdiri dari No., tanggal, bab, persentase, ttd dosen

Setelah selesai, susun dalam bentuk makalah hasil dari penggunaan query-query tsb.

DAFTAR PUSTAKA

- Coding, Secure. 2022. "MYSQL ATTACK."
<https://www.securecoding.com/blog/top-5-mysql-vulnerabilities-and-how-to-fix-them/>.
- Devart. "Types of SQL JOINS." 2022.
<https://www.devart.com/dbforge/sql/sqlcomplete/sql-join-statements.html>.
- DQLab. 2022. "Join Table." <https://dqlab.id/mengenal-macam-macam-fungsi-join-table-sql-dan-perbedaannya>.
- Encyclopaedia Britannica. 2022. "Android." *britannica.com*. 1.
- Oracle. 2023. "MySQL WorkBench."
<https://www.mysql.com/products/workbench/>.
- Point, Tutorial. 2022. "SQL - RIGHT JOINS."
<https://www.tutorialspoint.com/sql/sql-right-joins.htm>.
- Simplilearn. "Tutorial MySQL WorkBench."
<https://www.simplilearn.com/tutorials/mysql-tutorial/mysql-workbench-installation>.
- SoftwareTestingHelp. 2023. "MySQL Stored Procedure Parameters."
<https://www.mysqltutorial.org/stored-procedures-parameters.aspx>.

BIOGRAFI PENULIS



Gabriel Ardi Hutagalung, S.T., M. Kom, Penulis berkebangsaan Indonesia kelahiran 24 Juni 1992 di Kota Medan. Putera asli daerah berdarah batak toba dan Mandailing, anak sulung dari 3 bersaudara dengan almamater Magister Teknik Informatika Universitas Sumatera Utara Angkatan 2014. Lulus S1 Teknk Informatika dari Universitas Harapan medan pada Tahun 2014 dan lanjut ke Fakultas Ilmu Komputer Program Studi Teknik Informatika Universitas Sumatera Utara. Memulai karir sebagai seorang guru pada Sekolah Menengah Kejuruan Swasta pada Tahun 2015 sampai tahun 2018, Lulus Magister di tahun 2017 membawa penulis menjadi Pengajar di salah satu Perguruan tinggi di Kisaran Sumatera Utara. 3 semester menjadi dosen Sistem Informasi, penulis diberi amanah oleh negara untuk mengabdikan sebagai Dosen di Politeknik Negeri Medan pada tahun 2019 sampai saat ini.



Hikmah Adwin Adam, S.Kom., M. Kom, lahir di Medan pada tanggal 23 Maret 1970. Penulis menempuh Jenjang Pendidikan S1 di STMIK Budi Darma Medan. Pada tahun 2009 penulis melanjutkan kembali Jenjang Pendidikan S2 di Universitas Sumatera Utara. Mulai mengajar di Politeknik Negeri Medan pada tahun 2005.



Yuyun Yusnida Lase, S.Kom., M. Kom, lahir di Jakarta pada tanggal 06 Juli 1980. Pada tahun 1998 penulis menempuh Jenjang Pendidikan S1 di Universitas Putra Indonesia “YPTK” Padang. Pada tahun 2010 penulis melanjutkan kembali Jenjang Pendidikan S2 di Universitas Sumatera Utara. Awal berkarir pada tahun 2003 sebagai dosen swasta pada perguruan tinggi di kota Medan diantaranya Universitas Potensi Utama sampai tahun 2009, Universitas Budi Darma sampai tahun 2011 dan Universitas

Pembangunan Panca Budi sampai tahun 2014. Pada Tahun 2015 penulis mengajar di Politeknik Negeri Medan.



Santi Prayudani lahir di Kotamadya Binjai Sumatera Utara pada tanggal 28 Maret 1986. Penulis menempuh pendidikan S1 di Prodi Ilmu Komputer Universitas Sumatera Utara pada tahun 2004. Kemudian melanjutkan kembali pendidikan S2 di Prodi Teknik Informatika Universitas Sumatera Utara pada tahun 2011. Memulai karir sebagai guru di SDS Al Azhar Medan pada tahun 2010. Kemudian mengajar juga sebagai dosen di AMIK Harapan dan Universitas Pembangunan Panca Budi dari tahun 2011 sampai tahun 2014. Saat ini penulis diberi amanah oleh negara untuk mengabdikan sebagai dosen di Politeknik Negeri Medan dari tahun 2015.



Arif Ridho Lubis he got master from University Utara Malaysia in 2012 and graduate from University Utara Malaysia in 2011, both information technology. He is a lecturer in Department of Computer Engineering and Informatics, Politeknik Negeri Medan in 2015. His research interest includes computer science, network, science and project management. He can be contacted at email: arifridholubis@polmed.ac.id