

PRAKTIKUM MANAJEMEN BASIS DATA

PRODI SAINS DATA

IKOPIN UNIVERSITY



FITRI FATIMAH | KANAYA DZIKRA | 2025

KATA PENGANTAR

Database atau basis data adalah kumpulan data yang dikelola sedemikian rupa berdasarkan ketentuan tertentu yang saling berhubungan sehingga mudah dalam pengelolaannya. Melalui pengelolaan tersebut pengguna dapat memperoleh kemudahan dalam mencari informasi, menyimpan informasi dan membuang informasi. Basis Data (Database) dapat dibayangkan sebagai sebuah lemari arsip. Jika kita memiliki sebuah lemari arsip dan berwenang untuk mengelolanya.

Modul aplikasi basis data ini disusun sebagai bahan ajar bagi pembaca Program Studi Sains Data Universitas Koperasi Indonesia Fakultas Sains dan Teknologi, sehingga mampu mengenal dan mendalami bagaimana memahami serta mengimplementasikan pembuatan database. Pengembangan modul ini lebih kepada pengenalan serta praktek mengenai database yang nantinya akan dapat menjadi bahan rujukan bagi mahasiswa yang mendapatkan matakuliah ini.

Jatinangor, 5 Oktober 2025

Penulis

DAFTAR ISI

KATA PENGANTAR.....	i
DAFTAR ISI.....	ii
BAB I PENDAHULUAN.....	3
1.1 Pengenalan Phpmyadmin	3
1.2 Localhost/phpmyadmin	3
1.3 Instalasi Xampp	3
1.4 Akses phpMyAdmin.....	4
1.5 Latihan Praktikum	6
BAB II DESAIN BASIS DATA DI PHPMYADMIN	9
2.1 Membuat Tabel	9
2.2 Mengisi Record Pada Tabel	10
2.3 Mengubah dan Menghapus Isi Record	12
2.4 Mengubah dan Menghapus Field Struktur Tabel.....	12
2.5 Membuat Relasi Pada Tabel di Phpmyadmin	13
BAB III TUGAS.....	14
BAB IV REVIEW DAN LATIHAN	16
BAB V FUNGSI JOIN DAN MYSQL MARIADB	19
5.1 Inner Join	19
5.2 Msql.....	21
5.3 MariaDB	21
BAB VI AKSES COMMAND PROMPT DAN MARIADB.....	22
6.1 Command Prompt.....	22
6.2 Mengakses MariaDB dan Command Prompt.....	22
6.3 Contoh Pengerjaan Query.....	22

BAB I

PENDAHULUAN

1.1 Pengenalan Phpmyadmin

PhpMyAdmin merupakan tools dengan Graphic User Interface yang dapat memudahkan dalam pengelolaan database pada MySQL. phpMyAdmin adalah sebuah software yang berfungsi untuk mengelola MySQL yang ada di website. Dengan phpmyadmin dapat melakukan berbagai hal seperti membuat tabel, mengelola tabel, membuat kolom, melakukan indexing, mengelola hak akses user, menghapus data pada tabel, melakukan query dan sebagainya. Pada dasarnya operasi MySQL berupa konsol dan susah digunakan untuk pemula karena menggunakan query/ kode- kode yang beragam. Namun dengan hadirnya phpMyAdmin ini, semua orang dapat melakukan pengelolaan database dengan mudah. Anda dapat melakukan import database di phpMyAdmin juga.

Versi dari phpMyAdmin dan Namun, pada modul ini, pembahasan phpMyAdmin dibatasi hanya pada bab ini. Hal ini karena fokus utama adalah pemahaman dan penguasaan query SQL dalam membuat tabel dan basis data yang terstruktur dengan baik, sementara penggunaan phpMyAdmin hanya sebagai alat bantu untuk mempermudah praktik.

1.2 Localhost/phpmyadmin

Localhost/phpmyadmin adalah gabungan dari localhost dan phpmyadmin. phpmyadmin disimpan dalam host lokal di komputer yang disebut dengan localhost. Jadi intinya, localhost/phpmyadmin ini merupakan simulasi ketika user nantinya mengupload website di hosting yang sebenarnya. Hosting adalah tempat menyimpan semua file website. Sama seperti localhost, semua file website akan tersimpan dalam file lokal di komputer.

Untuk localhost/phpmyadmin, berfungsi untuk pengaturan pengelolaan database. Sehingga user tidak perlu menggunakan query manual secara konsol melalui mysql. User dapat melakukan proses insert table, memasukkan data secara mudah. Kesimpulannya adalah, localhost, phpMyAdmin dan XAMPP adalah kombinasi yang pas untuk membuat database di komputer sebelum diletakkan dalam hosting.

1.3 Instalasi Xampp

Xampp merupakan suatu tools yang bersifat open source yang sering dipergunakan untuk pengembangan aplikasi berbasis website yang didalamnya sudah menyediakan paket seperti Apache, MySQL, MariaDB, PHP, phpMyAdmin, FileZilla, Tomcat, Xampp Control Panel. Xampp sendiri telah tersedia juga untuk platform Windows mauppun Linux. Kita menggunakan Xampp versi 8.2.12

Berikut Langkah-langkah dalam penginstalan Xampp :

1. Kunjungi Situs Resmi XAMPP

Silakan buka situs resmi Apache Friends melalui tautan <https://www.apachefriends.org/index.html> , kemudian unduh XAMPP versi 8.2.12 sesuai dengan sistem operasi yang digunakan.

2. Proses Instalasi ke Local Disk C:

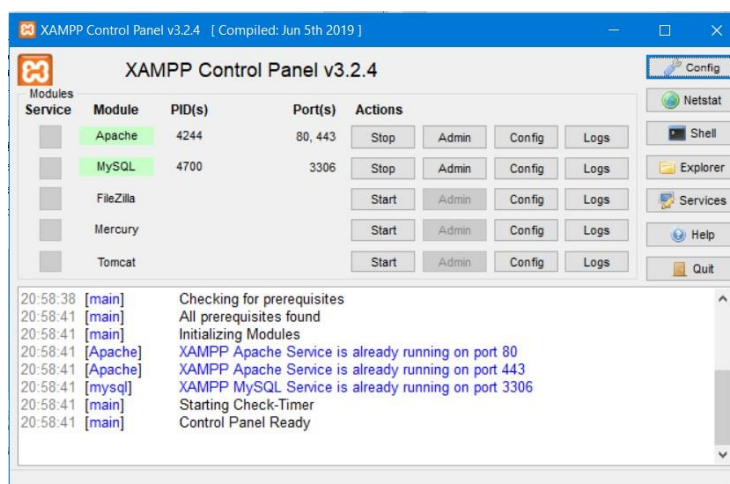
Jalankan file installer XAMPP yang telah diunduh. Pada saat proses instalasi, pilih lokasi penyimpanan di Local Disk C:\xampp agar mudah diakses saat pembelajaran. Setelah instalasi selesai, buka File Explorer, masuk ke lokasi C:\xampp, lalu cari file xampp-control.exe. Klik kanan pada file tersebut dan pilih Pin to Taskbar agar XAMPP Control Panel mudah dijalankan.

3. Menjalankan XAMPP Control Panel

Buka XAMPP Control Panel melalui taskbar, kemudian jalankan modul Apache dan MySQL dengan mengklik tombol Start. Pastikan kedua modul tersebut berstatus Running untuk menandakan bahwa server lokal telah aktif.

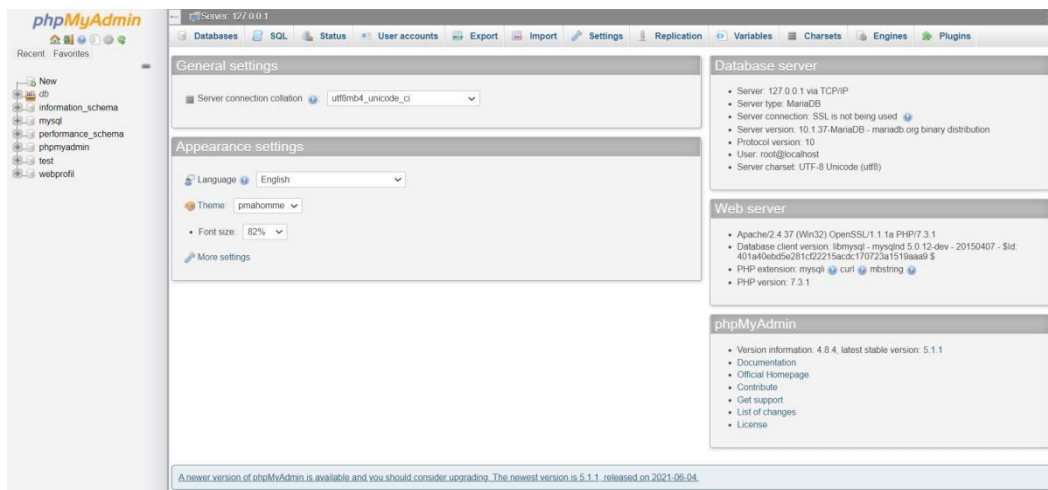
4. Pengujian Hasil Instalasi

Untuk memastikan XAMPP telah berfungsi dengan baik, klik tombol Admin pada modul MySQL di XAMPP Control Panel. Tindakan ini akan membuka halaman phpMyAdmin di browser secara otomatis. Apabila halaman phpMyAdmin berhasil muncul, maka instalasi XAMPP dinyatakan berhasil dan siap digunakan.



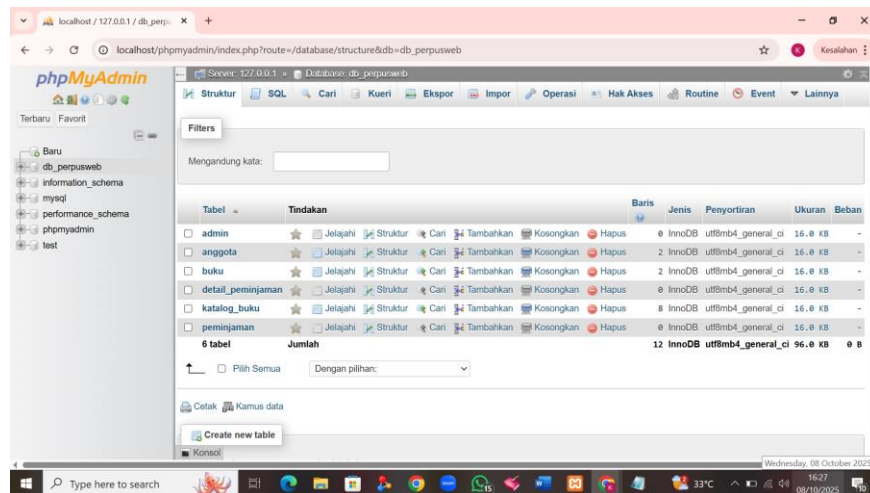
1.4 Akses phpMyAdmin

Setelah berhasil membuka phpMyAdmin, maka pengguna sudah bisa belajar phpMyAdmin dengan mencoba mengelola database MySQL, mulai dari membuat database baru sampai membuat tabel dan mengisinya. Pada bagian ini akan dijelaskan mengenai fitur yang ada di phpMyAdmin yang terintegrasi dengan cPanel. Bagian kanan terdapat menu Database server, berisi informasi mengenai server database. Sedangkan di bagian bawahnya terdapat Web server dan phpMyAdmin yang berisi informasi mengenai versi PHP dan phpMyAdmin. Untuk daftar database yang sudah dibuat akan terlihat pada bagian menu sebelah kiri. Dapat dilihat pada gambar d bawah ini.



Dalam phpMyAdmin, terdapat beberapa menu tambahan yang berfungsi untuk mengelola pengaturan tingkat lanjut pada database. Menu Basis Data (Databases) digunakan untuk membuat, menghapus, dan mengelola seluruh database yang ada di server, serta melihat daftar semua database beserta jumlah tabel yang dimilikinya. Menu SQL berfungsi untuk menjalankan perintah atau query secara langsung pada server database, seperti CREATE DATABASE, DROP TABLE, atau SELECT. Menu Status menampilkan informasi mengenai kondisi dan kinerja server MySQL, termasuk jumlah koneksi aktif, waktu berjalan, serta aktivitas query, sehingga pengguna dapat memantau performa server secara real-time. Menu Akun Pengguna (User Accounts) digunakan untuk mengatur hak akses dan izin bagi setiap pengguna database, termasuk menambah, menghapus, atau mengubah kata sandi serta menentukan hak akses terhadap database. Menu Ekspor (Export) berfungsi untuk menyalin data ke berbagai format file seperti SQL, CSV, PDF, atau XML, sedangkan Impor (Import) digunakan untuk memasukkan data dari file eksternal ke dalam database dengan format yang sama.

Menu Replikasi (Replication) digunakan untuk mengatur proses penyalinan data antar server database, yang berguna untuk membuat cadangan atau menyinkronkan data antar server. Menu Variabel (Variables) menampilkan daftar pengaturan internal MySQL seperti batas memori, ukuran buffer, dan konfigurasi koneksi, sehingga pengguna dapat meninjau atau menyesuaikan nilai variabel sesuai kebutuhan performa sistem. Menu Set Karakter (Character Set) digunakan untuk mengatur jenis pengkodean karakter (encoding) yang dipakai pada database, seperti utf8mb4 atau latin1, agar data teks tersimpan dan ditampilkan dengan benar. Menu Mesin (Engines) menampilkan daftar mesin penyimpanan (storage engines) yang tersedia, seperti InnoDB dan MyISAM, beserta karakteristik dan status penggunaannya. Terakhir, menu Plugin berfungsi untuk mengelola ekstensi atau modul tambahan, yang dapat diaktifkan, dinonaktifkan, atau ditinjau untuk memperluas kemampuan MySQL.

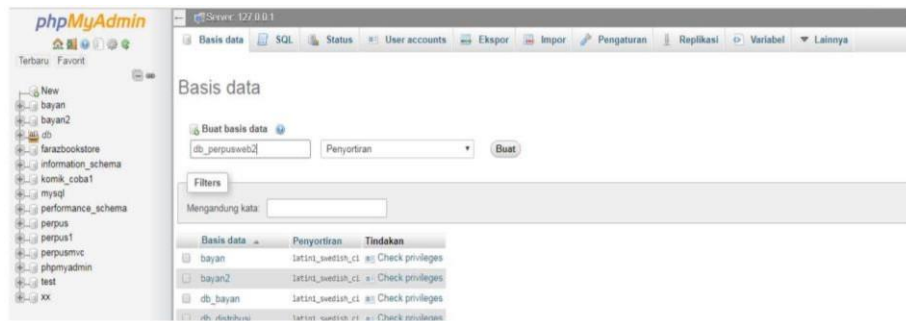


Pada phpMyAdmin juga, terdapat beberapa menu utama yang memiliki fungsi berbeda dalam pengelolaan database. Menu Structure digunakan untuk melihat dan mengatur struktur tabel seperti kolom, tipe data, serta kunci utama. Menu SQL berfungsi untuk menjalankan perintah atau query SQL secara langsung, seperti SELECT, INSERT, UPDATE, dan DELETE. Menu Search digunakan untuk mencari data atau nilai tertentu di dalam tabel database, sedangkan Query membantu pengguna menulis dan menjalankan kueri SQL dengan lebih mudah. Menu Export berfungsi untuk mengekspor database ke berbagai format seperti CSV, PDF, atau SQL, sementara Import digunakan untuk memasukkan data dari file luar dengan format serupa. Menu Operations menyediakan berbagai tindakan terhadap tabel seperti menggandakan, menghapus, atau mengubah nama tabel, sedangkan Triggers digunakan untuk membuat perintah otomatis yang dijalankan saat terjadi perubahan data, misalnya ketika data ditambah atau dihapus.

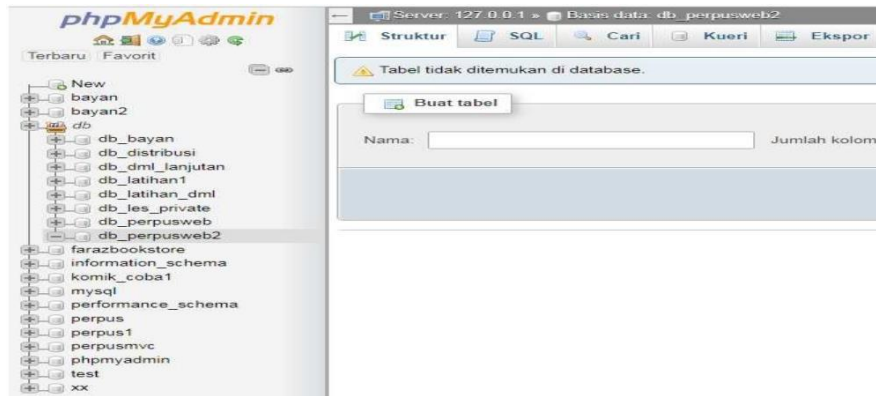
Selanjutnya, menu Routine berfungsi untuk membuat dan mengelola procedure atau function di dalam database, sedangkan Event digunakan untuk menjadwalkan tugas otomatis agar dijalankan pada waktu tertentu. Menu Pelacakan (Tracking) membantu memantau setiap perubahan pada struktur atau isi tabel, sementara Designer menampilkan diagram relasi antar tabel secara visual sehingga memudahkan dalam memahami hubungan data. Terakhir, Tengah Kolom (Central Columns) digunakan untuk mengatur kolom yang sering dipakai di beberapa tabel agar lebih konsisten dan mudah digunakan kembali.

1.5 Latihan Praktikum

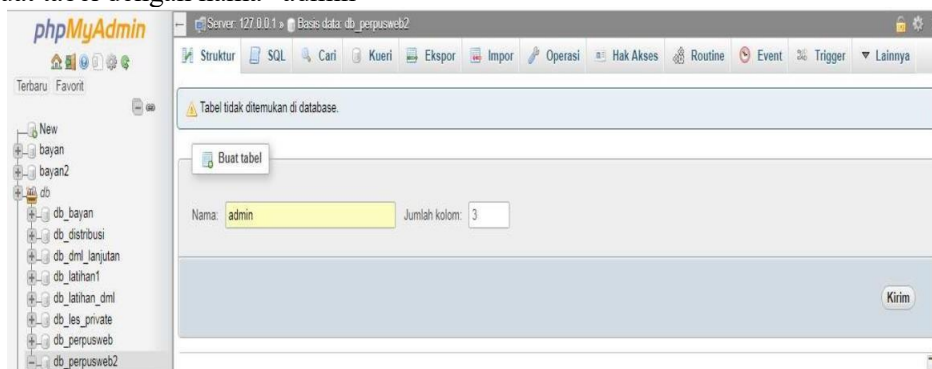
1. Membuat database
 - a. Ketikan pada kotak “Namadatabase” dibawah keterangan ‘Buat basis data’, lalu klik tombol “Buat”. Buat database dengan nama “db_perpusweb2”.



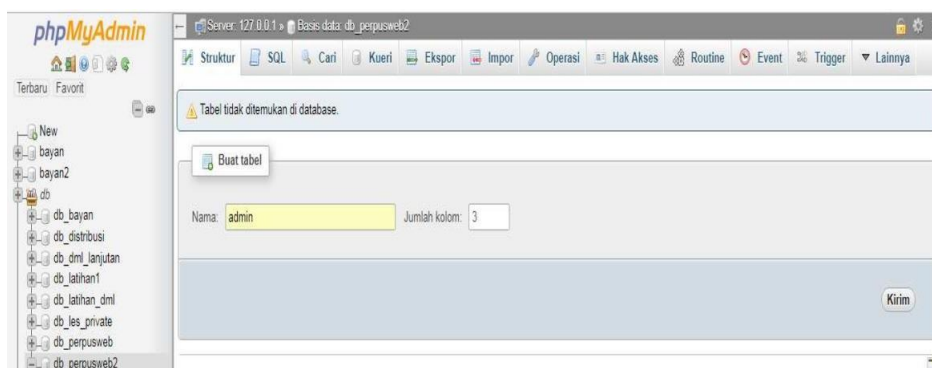
- b. Setelah membuat database baru, akan terlihat daftar tabel yang masih kosong (sisi sebelah kiri)



2. Membuat Tabel
 - Membuat tabel dengan nama “admin”



- Jika sudah mengisi nama tabel dan jumlah field klik tombol “Go”.



- Kemudian isikan data-data sebagai berikut

Server: 127.0.0.1 • Basis data: db_perpusweb2 • Tabel: admin

Nama tabel: admin column(s) (Kirim)

Nama	Jenis	Panjang/Nilai	Bawaan	Penyortiran	Atribut	Kosong	Indeks	A_I	Komenta
id_admin	INT	5	Tidak ada				PRIMARY	<input checked="" type="checkbox"/>	
username	VARCHAR	25	Tidak ada						
password	VARCHAR	35	Tidak ada						

Komentar tabel: Collation: Storage Engine: InnoDB

PARTITION definition: Partition by: Partitions:

Pratinjau SQL Simpan

Bila Primary Key, bersifat AUTO_INCREMENT, maka ceklis pada pilihan AUTO_INCREMENT atau A_I. Lalu Klik tombol Simpan. Tampilan ketika berhasil membuat tabel:

Server: 127.0.0.1 • Basis data: db_perpusweb2 • Tabel: admin

Struktur tabel Relation view

#	Nama	Jenis	Penyortiran	Atribut	Kosong	Bawaan	Komentar	Ekstra	Tindakan
1	id_admin	int(5)			Tidak	Tidak ada		AUTO_INCREMENT	Ubah Hapus Kunci Utama Unik Indeks Lainnya
2	username	varchar(25)	latin1_swedish_ci		Tidak	Tidak ada			Ubah Hapus Kunci Utama Unik Indeks Lainnya
3	password	varchar(35)	latin1_swedish_ci		Tidak	Tidak ada			Ubah Hapus Kunci Utama Unik Indeks Lainnya

Check all Dengan pilihan: Jelajahi Ubah Hapus Kunci Utama Unik Indeks Add to central columns Remove from central columns

Cetak Usulkan struktur tabel Lacak tabel Move columns Improve table structure

Tambahkan 1 kolom setelah password Kirim

Indeks

Tindakan	Nama kunci	Jenis	Unik	Dipadatkan	Kolom	Kardinalitas	Penyortiran	Kosong	Komentar
Ubah Hapus	PRIMARY	BTREE	Ya	Tidak	id_admin	0	A	Tidak	

Buat indeks pada 1 kolom Kirim

BAB II

DESAIN BASIS DATA DI PHPMYADMIN

2.1 Membuat Tabel

Buatlah beberapa tabel berdasarkan daftar tabel dan keterangan dibawah ini menggunakan tools phpmyadmin:

Tabel “**katalog_buku**”:

Field Name	Type	Size	Keterangan
Id_katalog	Int	5	Primary Key, AUTO_INCREMENT
Nama_katalog	Varchar	45	

Tabel “**buku**”:

Field Name	Type	Size	Keterangan
Id_buku	Int	5	Primary Key, AUTO_INCREMENT
Id_katalog	Int	5	
Judul_buku	Varchar	50	
Pengarang	Varchar	35	
Thn_terbit	Date	-	
penerbit	Varchar	50	

Tabel “**anggota**”:

Field Name	Type	Size	Keterangan
Id_anggota	Int	5	Primary Key, AUTO_INCREMENT
Nama	Varchar	45	
No_telp	Varchar	15	
Alamat	Varchar	50	
Email	Varchar	30	
Password	Varchar	35	

Tabel “**peminjaman**”:

Field Name	Type	Size	Keterangan
Id_pinjam	Int	5	Primary Key, AUTO_INCREMENT
Id_anggota	Int	5	
Tgl_pinjam	Date		
Tgl_kembali	Date		
Status	ENUM		‘Selesai’, ‘Belum Selesai’
Jml_buku	Int	2	

Tabel “**detail_peminjaman**”:

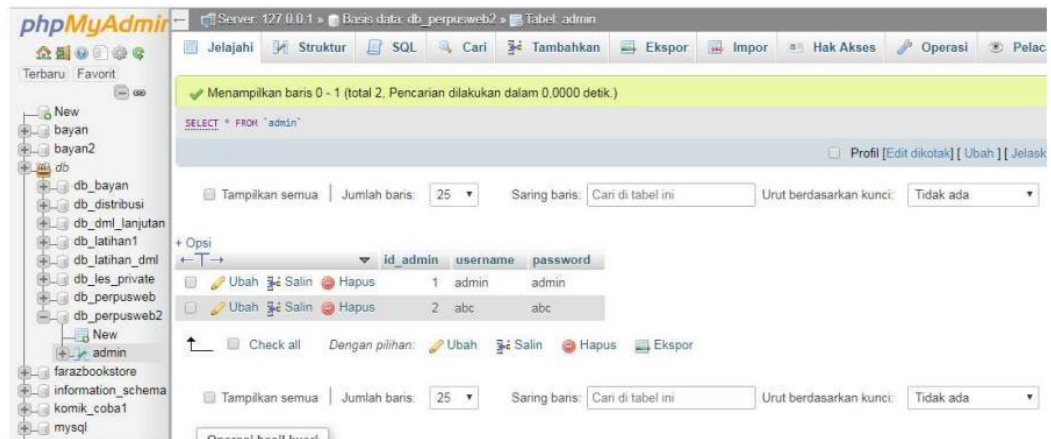
Field Name	Type	Size	Keterangan
Id_pinjam	Int	5	
Id_buku	Int	5	
Tgl_pengembalian	Date		
Denda	Double		
Status_buku	ENUM		‘Kembali’, ‘Belum Kembali’

2.2 Mengisi Record Pada Tabel

Untuk mengisi record tabel pada phpMyAdmin, klik/pilih terlebih dahulu tabel yang akan diisi recordnya pada daftar tabel sebelah kiri, kemudian klik menu tab “**Sisipkan/Tambahkan/Insert (Tambahkan)**”. Untuk Tabel dengan primary key bersifat AUTO_INCREMENT, maka saat insert record tidak perlu diisi data pada field primary key.

The screenshot shows the phpMyAdmin interface with the 'admin' table selected. The 'Tambahkan' (Insert) form is displayed, showing fields for 'id_admin', 'username', and 'password'. The 'id_admin' field is set to 'Kosong' (Empty). The 'username' field is set to 'admin' and the 'password' field is set to 'admin'. The 'Tambahkan sebagai baris baru' (Add as new row) dropdown is set to 'selanjutnya' (next). The 'Kirim' (Submit) button is visible.

Untuk menampilkan hasilnya klik menu tab “Browse / Jelajahi”



Latihan Lanjutan

Isilah record beberapa tabel yang telah dibuat berikut ini menggunakan phpMyAdmin. Tabel “**katalog_buku**”:

Id_katalog	Nama_katalog
1	Sains
2	Hobby
3	Komputer
4	Komunikasi
5	Hukum
6	Agama
7	Populer
8	Bahasa

Tabel “**buku**”:

Id_buku	Id_katalog	Judul_buku	Pengarang	Thn_terbit	Penerbit
1	1	Robotika Sederhana	Siswoyo Utomo	2013-01-01	Wacana Ria
2	3	Mahir dengan PHP	Adri Kusuma	2011-02-02	Pustaka Bangsa
3	2	Mahir Mewarnai	Akhmad Rahmat	2014-03-03	CV.Indo Kreasi
4	1	Hukum Fisika	Kurnia Sandi	2013-04-04	Wacana Ria

5	8	Mahir Bahasa Inggris	Aliuddin	2013-05-05	CV.Indo Kreasi
6	4	Public Speaking	Pambudi Prasetyo	2015-06-06	Aldi Pustaka
7	3	Trik SQL	Ahdim Makaren	2014-07-07	Wacana Ria
8	6	Kemurnian Agama	Pambudi Prasetyo	2014-08-08	Aldi Pustaka
9	1	Mikrokontroler	Ahdim Makaren	2012-09-09	Wacana Ria

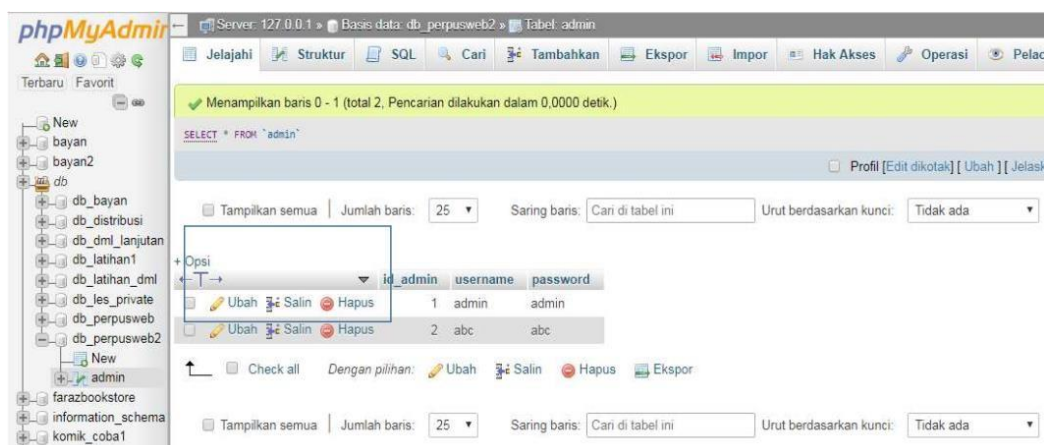
Tabel “anggota”:

Id_anggota	Nama	No_telp	Alamat	Email	password
1	Irfan Maulana	01244445555	BSD	irfan@gmail.com	123
2	Nur Kumalasari	01233335555	Ciledug	nur@gmail.com	123
3	Sanjaya Wijaya	01211115555	Cimone	sanjaya@gmail.com	123
4	Eva Irfianingsih	01266665555	Tangerang	eva@gmail.com	123
5	Ifqoh Permatasari	01277775555	Cengkareng	ifqoh@gmail.com	123
6	Indah Riana	01288885555	Fatmawati	indah@gmail.com	123

7	Tiwie Andrawati	01299995555	Warung Jati	tiwie@gmail.com	123
8	Mus Dalifa	01200005555	Jatiwaringin	mus@gmail.com	123
9	Hisbu Utomo	01233336666	Salemba	hisbu@gmail.com	123
10	Zaenal Abidin	01233337777	Bekasi	zaenal@gmail.com	123

2.3 Mengubah dan Menghapus Isi Record

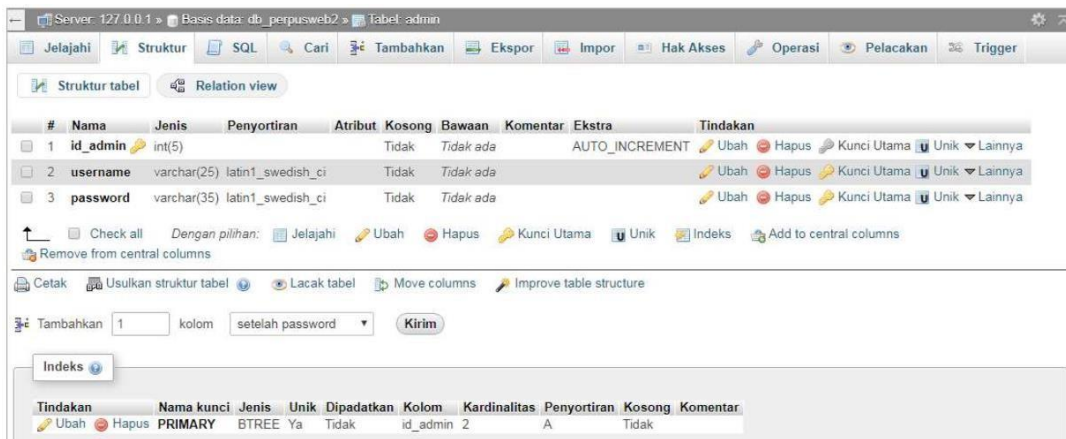
Pilih terlebih dahulu tabel, kemudian klik menu “**Browse / Jelajahi**” untuk menampilkan isi record pada tabel. Untuk Ubah bisa menggunakan icon pensil, sedangkan untuk Hapus bisa menggunakan icon tanda silang/strip merah.



2.4 Mengubah dan Menghapus Field Struktur Tabel

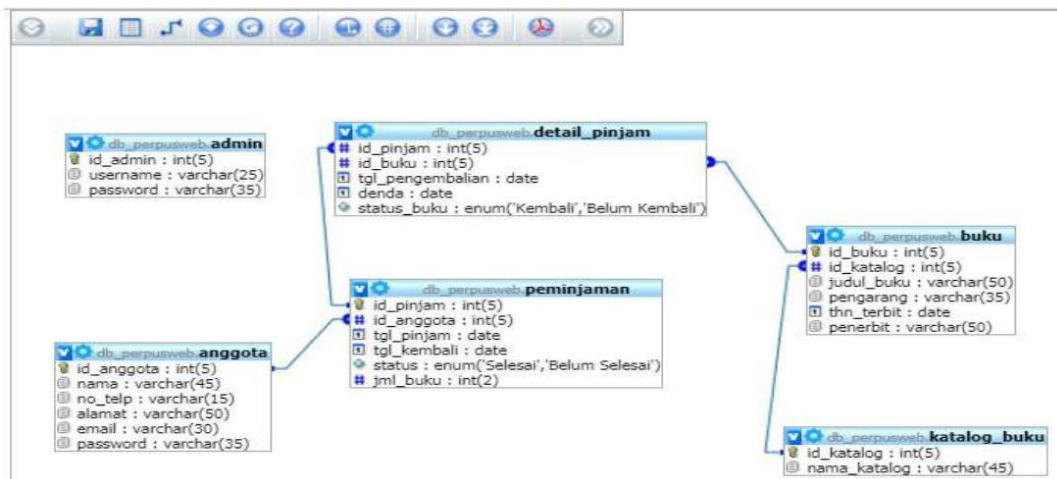
Pilih terlebih dahulu tabel, kemudian klik menu “Struktur” untuk menampilkan isi record pada tabel. Untuk Ubah bisa menggunakan icon pensil, memberi Primary Key menggunakan icon

kunci, sedangkan untuk Hapus bisa menggunakan icon tanda silang/strip merah.



2.5 Membuat Relasi Pada Tabel di Phpmyadmin

Pilih dan klik menu tab “**Designer**” Saat Menu Designer terbuka, secara default field- field pada database belum terbuka. Cara membuka field pada tabel klik tanda segitiga pada masing-masing kotak tabel. Kemudian buat relasi dengan mengklik icon “Create Relation”, relasikan field primary key pada masing-masing tabel ke tabel lain dengan field yang sama dengan teknik “drag and drop”.



BAB III

TUGAS

Berikut latihan pembuatan database menggunakan phpmyadmin dengan ketentuan pada tabel di bawah ini:

1. Buat database menggunakan phpMyAdmin dengan nama “db_latihan_sepatu”.
2. Buat beberapa tabel pada database db_latihan_sepatu, sebagai berikut: Tabel

“sepatu”:

Field Name	Type	Size	Keterangan
Kd_sepatu	Char	4	Primary Key
merk	Varchar	30	
Ukuran	Varchar	10	
Harga	Int	8	

Tabel “pembeli”:

Field Name	Type	Size	Keterangan
Id_pembeli	Int	5	Primary Key, AUTO_INCREMENT
Nm_pembeli	Varchar	35	
Alamat	Varchar	60	
No_hp	Varchar	18	

Tabel “penjualan”:

Field Name	Type	Size	Keterangan
No_fak	Char	6	Primary Key
Tgl_fak	Varchar	35	
Id_pembeli	Int	5	

Tabel “detail_penjualan”:

Field Name	Type	Size	Keterangan
No_fak	Char	6	
Kd_sepatu	Char	4	
Jumlah	Int	3	

3. Kemudian isi record tabel-tabel yang telah dibuat

Tabel “sepatu”:

Kd_sepatu	merk	Ukuran	Harga
S001	Nike	40	300.000
S002	Adidas	41	320.000
S003	Xander	40	350.000
S004	Mutiara	40	240.000
S005	Bata	41	280.000
S006	Penx	41	230.000

Tabel “pembeli”:

id_pembeli	Nm_pembeli	Alamat	No_hp
1	Joko	Pontianak	091212123434
2	Butet	Yogyakarta	091213134545
3	Daeng	Tegal	091214145656
4	Putu	Tasikmalaya	091215156767
5	Neng	Bandung	091216167878

Tabel “penjualan”:

No_fak	Tgl_fak	Id_pembeli
FK0101	2017-11-02	1
FK0102	2017-11-03	3
FK0103	2017-11-04	4
FK0104	2017-11-05	5

Tabel “detail_penjualan”:

No_fak	Kd_sepatu	Jumlah
FK0101	S001	1
FK0101	S002	2
FK0102	S001	1
FK0103	S005	3
FK0104	S003	1
FK0104	S006	1

4. Buat relasi antar tabel dengan menggunakan “Designer phpMyAdmin”.

BAB IV
REVIEW DAN LATIHAN

1. Buat Database

Nama database: **db_kursus_online**

2. Buat tabel-tabel berikut di phpMyAdmin

Tabel 'instruktur'

Nama Field	Tipe Data	Keterangan
id_instruktur	INT(11)	PRIMARY KEY, AUTO_INCREMENT
nama_instruktur	VARCHAR(30)	
email	VARCHAR(30)	
no_hp	VARCHAR(20)	
keahlian	VARCHAR(30)	

Tabel 'kursus'

Nama Field	Tipe Data	Keterangan
id_kursus	INT(11)	PRIMARY KEY, AUTO_INCREMENT
nama_kursus	VARCHAR(30)	
kategori	VARCHAR(2)	
durasi_jam	INT(5)	
id_instruktur	INT(11)	FOREIGN KEY

Tabel 'peserta'

Nama Field	Tipe Data	Keterangan
id_peserta	INT(11)	PRIMARY KEY, AUTO_INCREMENT
nama_peserta	VARCHAR(30)	
email	VARCHAR(30)	
no_hp	VARCHAR(20)	
alamat	VARCHAR(30)	

Tabel 'pendaftaran'

Nama Field	Tipe Data	Keterangan
id_pendaftaran	INT(11)	PRIMARY KEY, AUTO_INCREMENT
id_peserta	INT(11)	FOREIGN KEY >peserta(id_peserta)
Id_pembayaran	INT(11)	FOREIGN KEY>pembayaran(id_pembayaran)
id_kursus	INT(11)	FOREIGN KEY >kursus(id_kursus)
tanggal_daftar	DATE	
status	ENUM('aktif','selesai','batal')	

Tabel 'pembayaran'

Nama Field	Tipe Data	Keterangan
id_pembayaran	INT(11)	PRIMARY KEY, AUTO_INCREMENT
id_pendaftaran	INT(11)	FOREIGN KEY >pendaftaran(id_pendaftaran)
tanggal_bayar	DATE	
jumlah_bayar	DECIMAL(10,2)	
metode_bayar	ENUM('transfer','qris','cash')	

Isi Data (Record)

Tabel 'instruktur'

id_instruktur	nama_instruktur	email	no_hp	keahlian
1	Rina Setiawan	rina@kampus.ac.id	08967555276889	Data Science
2	Dimas Fadillah	dimas@kampus.ac.id	089673768643863	UI/UX Design
3	Tia Pramesti	tia@kampus.ac.id	089653865829028	Web Development

Tabel 'kursus'

id_kursus	nama_kursus	kategori	durasi_jam	id_instruktur
101	Python untuk Pemula	Pemrograman	20	1
102	Desain Antarmuka Dasar	Desain	15	2
103	Website dengan HTML & CSS	Pemrograman	25	3

Tabel 'peserta'

id_peserta	nama_peserta	email	no_hp
501	Budi Santoso	budi@student.univ.ac.id	089876543210
502	Nanda Putri	nanda@student.univ.ac.id	081234998877
503	Ali Rahman	ali@student.univ.ac.id	082134567890

Tabel 'pendaftaran'

id_pendaftaran	id_peserta	id_pembayaran	id_kursus	tanggal_daftar	status
9001	501	7001	101	2025-03-10	aktif
9002	502	7002	102	2025-03-15	selesai
9003	503	7003	103	2025-03-20	aktif

Tabel 'pembayaran'

id_pembayaran	id_pendaftaran	tanggal_bayar	jumlah_bayar	metode_bayar
7001	9001	2025-03-11	250000.00	qris
7002	9002	2025-03-16	200000.00	transfer
7003	9003	2025-03-21	300000.00	cash

BAB V

FUNGSI JOIN DAN MYSQL MARIADB

5.1 Inner Join

Join merupakan suatu perintah atau query yang digunakan untuk mendapatkan atau menampilkan data yang berasal dari dua buah tabel atau lebih. Syarat dari terpenuhinya query join adalah tabel-tabel yang digunakan dalam query harus memiliki relasi satu sama lain. (*lihat bahasan studi kasus perpustakaan*). Secara umum, query join yang sering digunakan terdiri dari **Inner Join** dan **Outer Join (Left dan Right)**.

Inner Join merupakan perintah untuk menampilkan semua data yang mempunyai nilai sama, kita bisa lihat pada perintah di bawah ini, **ON rsh_motor.id_brand = rsh_brand.id** yang artinya data yang ditampilkan adalah data yang sama dan berhubungan antara **id_brand** yang ada pada tabel **rsh_motor** dengan id pada tabel **rsh_brand**.

Tabel: rsh_motor

id	motor	id_brand
1	Ninja 250	1
2	CBR 250 R	2
3	NMAX 250	3
4	Vixion 150	3
5	Satria F 150	5
6	Pulsar	6

Tabel: rsh_brand

id	brand
1	Kawasaki
2	Honda
3	Yamaha
4	Bajaj
5	Suzuki

Berikut contoh perintah tersebut:

```
SELECT * FROM rsh_motor INNER JOIN rsh_brand
```

```
ON rsh_motor.id_brand = rsh_brand.id
```

Hasil dari perintah tersebut:

id	motor	id_brand	id	brand
1	Ninja 250	1	1	Kawasaki
2	CBR 250 R	2	2	Honda
3	NMAX 250	3	3	Yamaha
4	Vixion 150	3	3	Yamaha
5	Satria F 150	5	5	Suzuki

Sedangkan Left outer join (left join) merupakan cara menghubungkan tabel dan menampilkan semua data (kiri) pada tabel yang tidak berhubungan, sedangkan data yang kosong akan bernilai NULL. Untuk Right outer Join, fungsi ini hampir sama dengan fungsi Left outer Join ataupun kebalikannya, dimana Right Join akan menghubungkan tabel dan menampilkan semua data (kanan) pada tabel yang tidak berhubungan, dan data yang kosong akan bernilai NULL.

Contoh perintah SQL Left Outer Join:

SELECT * FROM rsh_motor LEFT JOIN rsh_brand

ON rsh_motor.id_brand = rsh_brand.id

Hasil dari perintah tersebut:

id	motor	id_brand	id	brand
1	Ninja 250	1	1	Kawasaki
2	CBR 250 R	2	2	Honda
3	NMAX 250	3	3	Yamaha
4	Vixion 150	3	3	Yamaha
5	Satria F 150	5	5	Suzuki
6	Pulsar	6	NULL	NULL

Contoh perintah SQL Right Outer Join:

SELECT * FROM rsh_motor RIGHT JOIN rsh_brand

ON rsh_motor.id_brand = rsh_brand.id

Hasil dari perintah tersebut:

id	motor	id_brand	id	brand
1	Ninja 250	1	1	Kawasaki
2	CBR 250 R	2	2	Honda
3	NMAX 250	3	3	Yamaha
4	Vixion 150	3	3	Yamaha
5	Satria F 150	5	5	Suzuki
NULL	NULL	NULL	4	Bajaj

5.2 Mysql

MySQL adalah sebuah program database server yang mampu menerima dan mengirimkan datanya dengan sangat cepat, multi user serta menggunakan perintah standar SQL (Structure Query Language). MySQL dan SQL adalah dua software yang berbeda. MySQL merupakan salah satu nama brand terpopuler dari software RDBMS yang menerapkan client server model. Keduanya menggunakan bahasa spesifik domain Structured Query Language (SQL). Software RDBMS selalu menggunakan SQL sebagai bahasa utama untuk berinteraksi dengan database. MySQL sendiri ditulis dalam C dan C++. Tugas SQL adalah untuk memberitahukan server tentang apa yang harus dilakukannya terhadap data. Penggambaran umumnya seperti password atau kode WordPress. Ketika kita memasukkan password atau kode tersebut ke sistem untuk mendapatkan akses agar bisa login ke dashboard.

Dalam hal ini, SQL statement menginstruksikan server untuk menjalankan operasi sebagai berikut:

- Data query: meminta informasi yang spesifik dari database yang sudah ada.
- Manipulasi data: menambahkan, menghapus, mengubah, menyortir, melakukan operasi lainnya untuk memodifikasi data, value, atau visual.
- Identitas data (data identity): menentukan tipe data, misalnya mengubah data numerik menjadi data integer. Selain itu, juga menentukan schema atau hubungan dari masing masing tabel yang ada di database.
- Data access control: menyediakan metode keamanan untuk melindungi data, termasuk dalam menentukan siapa yang boleh melihat atau menggunakan informasi yang tersimpan di database.

5.3 MariaDB

Seiring akuisisi dan dimilikinya MySQL oleh Oracle, maka para pengembang awal MySQL membangun sebuah sistem manajemen database baru yang dinamakan MariaDB. MariaDB memiliki SQL dalam DNA-nya dan memungkinkan pemrosesan data gaya ACID dengan atomisitas terjamin, daya tahan, isolasi, dan konsistensi untuk transaksi. Fitur Utama MariaDB adalah include support API JSON, replikasi data paralel dan berbagai mesin penyimpanan.

Tetapi ada beberapa area yang tidak kompatibel antar versi. Misalnya, dari versi 10.1 ke atas, MariaDB menyimpan data JSON menggunakan format berbeda ke MySQL 5.7. Untuk menyiasatinya, bagi yang ingin menyalin kolom objek JSON dari MySQL ke MariaDB dapat mengkonversikannya ke format yang menggunakan atau menjalankan pekerjaan replikasi berbasis pernyataan dengan SQL. MariaDB memiliki fungsi basis data kolom yang dirancang untuk lebih mendukung analitik waktu nyata dalam skala besar. Sehingga bisa dikatakan MariaDB merupakan open source pengembangan dari MySQL yang full free. Xampp sendiri telah mengganti database MySQL dengan MariaDB sejak versi 5.5.3. Sehingga modul praktikum ini menggunakan basis data dengan Jenis Server MariaDB.

BAB VI

AKSES COMMAND PROMPT DAN MARIADB

6.1 Command Prompt

Command Prompt adalah aplikasi command line interpreter yang tersedia di sebagian besar sistem operasi Windows. Ini digunakan untuk menjalankan perintah yang dimasukkan. Sebagian besar dari perintah itu mengotomatiskan tugas-tugas melalui skrip dan batch files, melakukan fungsi administrasi tingkat lanjut, dan memecahkan masalah atau memecahkan beberapa jenis masalah Windows. Untuk membuka command prompt pada Laptop/PC bisa menggunakan berbagai cara, berikut salah satu caranya:

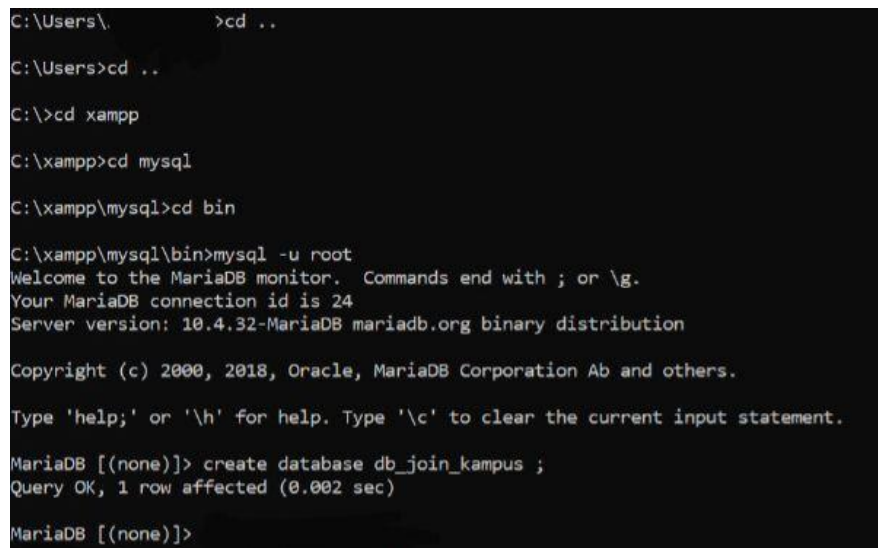
- a. Pilih menu Start - All Programs - Accessories Kemudian Pilih Command Prompt.
- b. Klik Kanan pada command prompt atau cmd, lalu pilih Run as administrator.

6.2 Mengakses MariaDB dan Command Prompt

1. Ketikan perintah `cd c:\` (lalu kemudian tekan Enter)
2. Lalu ketikan perintah `"xampp/mysql/bin/mysql.exe" -u root -p` (lalu kemudian tekan Enter)

Latihan Praktikum MariaDB:

1. Masuk kedalam MariaDB melalui command prompt.
2. Buat sebuah database dengan nama `"db_join_kampus"`, kemudian tampilkan seluruh daftar tabel yang ada.
3. Aktifkan database `db_join_kampus`.



```
C:\Users\>cd ..
C:\Users>cd ..
C:\>cd xampp
C:\xampp>cd mysql
C:\xampp\mysql>cd bin
C:\xampp\mysql\bin>mysql -u root
Welcome to the MariaDB monitor.  Commands end with ; or \g.
Your MariaDB connection id is 24
Server version: 10.4.32-MariaDB mariadb.org binary distribution

Copyright (c) 2000, 2018, Oracle, MariaDB Corporation Ab and others.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

MariaDB [(none)]> create database db_join_kampus ;
Query OK, 1 row affected (0.002 sec)

MariaDB [(none)]>
```

4. Buat beberapa tabel sebagai berikut: Tabel **"dosen"**:

Field Name	Type	Size	Keterangan
Nip	Char	5	Primary Key
Nm_dosen	Varchar	35	
Thn_masuk	Year	-	
Alamat	Varchar	50	

```

MariaDB [db_join_kampus]> CREATE TABLE dosen (
  -> nip CHAR(5) NOT NULL,
  -> nm_dosen VARCHAR(35) NOT NULL,
  -> thn_masuk YEAR NOT NULL,
  -> alamat VARCHAR(50) NOT NULL,
  -> PRIMARY KEY (nip)
  -> );
Query OK, 0 rows affected (0.016 sec)

```

Tabel “mt_kuliah”:

Field Name	Type	Size	Keterangan
kd_mtkul	Char	3	Primary Key
Nm_mtkul	Varchar	30	
Sks	Int	2	

```

MariaDB [db_join_kampus]> CREATE TABLE dosenmt_kuliah (
  -> kd_mtkul CHAR(3) NOT NULL,
  -> nm_mtkul VARCHAR(30) NOT NULL,
  -> sks INT(2) NOT NULL,
  -> PRIMARY KEY (kd_mtkul)
  -> );
Query OK, 0 rows affected (0.013 sec)

MariaDB [db_join_kampus]>

```

Tabel “mengajar”:

Field Name	Type	Size	Keterangan
Id	Int	3	Primary Key, AUTO_INCREMENT
Nip	Char	5	Foreign Key
kd_mtkul	Char	3	Foreign Key

```

MariaDB [db_join_kampus]> CREATE TABLE mengajar (
-> id INT(3) NOT NULL AUTO_INCREMENT,
-> nip CHAR(5) NOT NULL,
-> kd_matkul CHAR(3) NOT NULL,
-> PRIMARY KEY (id)
-> );
Query OK, 0 rows affected (0.015 sec)

MariaDB [db_join_kampus]>

```

5. Tampilkan struktur tabel dari masing-masing tabel yang telah dibuat.

```

MariaDB [db_join_kampus]> DESC dosen; DESC dosenmt_kuliah; DESC mengajar;
+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| nip   | char(5) | NO | PRI | NULL | |
| nm_dosen | varchar(35) | NO | | NULL | |
| thn_masuk | year(4) | NO | | NULL | |
| alamat | varchar(50) | NO | | NULL | |
+-----+-----+-----+-----+-----+-----+
4 rows in set (0.034 sec)

+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| kd_mtkul | char(3) | NO | PRI | NULL | |
| nm_mtkul | varchar(30) | NO | | NULL | |
| sks     | int(2) | NO | | NULL | |
+-----+-----+-----+-----+-----+-----+
3 rows in set (0.005 sec)

+-----+-----+-----+-----+-----+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| id     | int(3) | NO | PRI | NULL | auto_increment |
| nip    | char(5) | NO | | NULL | |
| kd_matkul | char(3) | NO | | NULL | |
+-----+-----+-----+-----+-----+-----+
3 rows in set (0.004 sec)

MariaDB [db_join_kampus]>

```

6. Isikan data untuk tabel dosen, dengan data sebagai berikut: Tabel “**dosen**”:

NIP	Nm_dosen	Thn_masuk	Alamat
01011	Budi Santoso	2003	Tangerang
01012	Rio Hermawan	2009	Jakarta
01013	Tio Sandewa	2005	Jakarta
01014	Agus Supardi	2011	Bogor
01015	Nurlaela Sari	2014	Depok

```
MariaDB [db_join_kampus]> INSERT INTO dosen (nip, nm_dosen, thn_masuk, alamat)
-> VALUES
-> ('01011', 'Budi Santoso', 2003, 'Tangerang'),
-> ('01012', 'Rio Hermawan', 2009, 'Jakarta'),
-> ('01013', 'Tio Sandewa', 2005, 'Jakarta'),
-> ('01014', 'Agus Supandi', 2011, 'Bogor'),
-> ('01015', 'Nurlaela Sari', 2014, 'Depok');
Query OK, 5 rows affected (0.006 sec)
Records: 5 Duplicates: 0 Warnings: 0
MariaDB [db_join_kampus]>
```

Tabel “**mt_kuliah**”:

kd_mtkul	Nm_mtkul	Sks
101	Algoritma	4
202	PTIK	3
303	SIM	4
404	E-Commerce	2
505	Agama	3

```

MariaDB [db_join_kampus]> INSERT INTO dosenmt_kuliah (kd_mtkul, nm_mtkul, sks)
-> VALUES
-> ('101', 'Algoritma', 4),
-> ('202', 'PTIK', 3),
-> ('303', 'SIM', 4),
-> ('404', 'E-Commerce', 2),
-> ('505', 'Agama', 3);
Query OK, 5 rows affected (0.005 sec)
Records: 5 Duplicates: 0 Warnings: 0

MariaDB [db_join_kampus]>

```

Tabel “mengajar”:

Id	Nip	kd_mtkul
1	01011	101
2	01011	303
3	01012	202
4	01012	404
5	01012	303
6	01013	505
7	01013	101
8	01014	404
9	01015	202
10	01015	303

```

MariaDB [db_join_kampus]> INSERT INTO mengajar (nip, kd_matkul)
-> VALUES
-> ('01011', '101'),
-> ('01011', '303'),
-> ('01012', '202'),
-> ('01012', '404'),
-> ('01012', '303'),
-> ('01013', '505'),
-> ('01013', '101'),
-> ('01014', '404'),
-> ('01015', '202'),
-> ('01015', '303');
Query OK, 10 rows affected (0.005 sec)
Records: 10 Duplicates: 0 Warnings: 0

MariaDB [db_join_kampus]>

```

7. Tampilkan isi record dari ketiga tabel yang telah dibuat. (dengan perintah Select)

```
MariaDB [db_join_kampus]> SELECT * FROM dosen;
```

nip	nm_dosen	thn_masuk	alamat
01011	Budi Santoso	2003	Tangerang
01012	Rio Hermawan	2009	Jakarta
01013	Tio Sandewa	2005	Jakarta
01014	Agus Supandi	2011	Bogor
01015	Nurlaela Sari	2014	Depok

```
5 rows in set (0.001 sec)
```

```
MariaDB [db_join_kampus]> SELECT * FROM dosenmt_kuliah;
```

kd_mtkul	nm_mtkul	sks
101	Algoritma	4
202	PTIK	3
303	SIM	4
404	E-Commerce	2
505	Agama	3

```
5 rows in set (0.000 sec)
```

```
MariaDB [db_join_kampus]> SELECT * FROM mengajar;
```

id	nip	kd_mtkul
1	01011	101
2	01011	303
3	01012	202
4	01012	404
5	01012	303
6	01013	505
7	01013	101
8	01014	404
9	01015	202
10	01015	303

```
10 rows in set (0.000 sec)
```

8. Tampilkan daftar dosen yang mengajar mata kuliah = "SIM" menggunakan Inner Join.

```
MariaDB [db_join_kampus]> SELECT d.nip, d.nm_dosen, m.nm_mtkul
-> FROM dosen d
-> INNER JOIN mengajar mg ON d.nip = mg.nip
-> INNER JOIN dosenmt_kuliah m ON mg.kd_mtkul = m.kd_mtkul
-> WHERE m.nm_mtkul = 'SIM';
```

nip	nm_dosen	nm_mtkul
01011	Budi Santoso	SIM
01012	Rio Hermawan	SIM
01015	Nurlaela Sari	SIM

```
3 rows in set (0.001 sec)
```

```
MariaDB [db_join_kampus]>
```

9. Tampilkan hanya field "nm_dosen", "nm_mtkul", "thn_masuk" yang mengajar mata kuliah = "E-Commerce", kemudian urutkan secara descending berdasarkan "thn_masuk".

```
MariaDB [db_join_kampus]> SELECT d.nm_dosen, m.nm_mtkul, d.thn_masuk
-> FROM dosen d
-> INNER JOIN mengajar mg ON d.nip = mg.nip
-> INNER JOIN dosenmt_kuliah m ON mg.kd_mtkul = m.kd_mtkul
-> WHERE m.nm_mtkul = 'E-Commerce'
-> ORDER BY d.thn_masuk DESC;
```

nm_dosen	nm_mtkul	thn_masuk
Agus Supandi	E-Commerce	2011
Rio Hermawan	E-Commerce	2009

```
2 rows in set (0.002 sec)
```

```
MariaDB [db_join_kampus]>
```

10. Buat alias dengan nama "Nama Dosen" untuk field "nm_dosen" dan alias "Mengajar" untuk field "nm_mtkul". Tampilkan hanya field NIP, alias Nama Dosen, alias Mengajar, kemudian urutkan berdasarkan "nm_mtkul".

```
MariaDB [db_join_kampus]> SELECT
-> d.nip,
-> d.nm_dosen AS 'Nama Dosen',
-> m.nm_mtkul AS 'Mengajar'
-> FROM dosen d
-> INNER JOIN mengajar mg ON d.nip = mg.nip
-> INNER JOIN dosenmt_kuliah m ON mg.kd_mtkul = m.kd_mtkul
-> ORDER BY m.nm_mtkul;
```

nip	Nama Dosen	Mengajar
01013	Tio Sandewa	Agama
01013	Tio Sandewa	Algoritma
01011	Budi Santoso	Algoritma
01014	Agus Supandi	E-Commerce
01012	Rio Hermawan	E-Commerce
01015	Nurlaela Sari	PTIK
01012	Rio Hermawan	PTIK
01012	Rio Hermawan	SIM
01011	Budi Santoso	SIM
01015	Nurlaela Sari	SIM

```
10 rows in set (0.001 sec)
```

```
MariaDB [db_join_kampus]>
```

11. Membuka Database yang Sudah Ada

Langkah ini dilakukan apabila database *db_join_kampus* sudah pernah dibuat sebelumnya dan kita ingin menggunakannya kembali tanpa membuat ulang.

Langkah-langkah:

- Tampilkan semua database yang ada di sistem menggunakan perintah berikut:

```
MariaDB [db_join_kampus]> SHOW DATABASES;
+-----+
| Database |
+-----+
| db_join_kampus |
| db_kursus_online |
| db_latihan_sepatu |
| db_perpusweb |
| information_schema |
| mysql |
| performance_schema |
| phpmyadmin |
| test |
+-----+
9 rows in set (0.002 sec)
```

Dari hasil di atas, terlihat bahwa database *db_join_kampus* sudah ada.

- Pilih dan gunakan database tersebut dengan perintah:

```
MariaDB [db_join_kampus]> USE db_join_kampus;
Database changed
MariaDB [db_join_kampus]> _
```

- Lihat semua tabel yang ada di dalam database dengan perintah:

```
MariaDB [db_join_kampus]> SHOW TABLES;
+-----+
| Tables_in_db_join_kampus |
+-----+
| dosen |
| dosenmt_kuliah |
| mengajar |
| mt_kuliah |
+-----+
4 rows in set (0.002 sec)
```

- Periksa struktur tabel jika ingin mengetahui nama kolom dan tipe datanya:

```
MariaDB [db_join_kampus]> DESC dosen;
+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+
| nip | char(5) | NO | PRI | NULL | |
| nm_dosen | varchar(35) | NO | | NULL | |
| thn_masuk | year(4) | NO | | NULL | |
| alamat | varchar(50) | NO | | NULL | |
+-----+
4 rows in set (0.027 sec)

MariaDB [db_join_kampus]> DESC mt_kuliah;
+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+
| kd_mtkul | char(3) | NO | PRI | NULL | |
| nm_mtkul | varchar(30) | NO | | NULL | |
| sks | int(1) | NO | | NULL | |
+-----+
3 rows in set (0.003 sec)

MariaDB [db_join_kampus]> DESC mengajar;
+-----+
| Field | Type | Null | Key | Default | Extra |
+-----+
| id | int(3) | NO | PRI | NULL | auto_increment |
| nip | char(5) | NO | | NULL | |
| kd_mtkul | char(3) | NO | | NULL | |
+-----+
3 rows in set (0.003 sec)
```

Dengan langkah di atas, pengguna dapat membuka kembali database yang sudah ada tanpa harus membuat ulang, serta memastikan tabel dan struktur data di dalamnya masih tersimpan dengan benar.

6.3 Contoh Pengerjaan Query

- a. Menampilkan daftar dosen beserta mata kuliah yang diajarkan

```
MariaDB [db_join_kampus]> SELECT d.nm_dosen AS 'Nama Dosen', m.nm_mtkul AS 'Mengajar'
-> FROM dosen d
-> JOIN mengajar mg ON d.nip = mg.nip
-> JOIN mt_kuliah m ON mg.kd_mtkul = m.kd_mtkul;

+-----+-----+
| Nama Dosen | Mengajar |
+-----+-----+
| Budi Santoso | Algoritma |
| Budi Santoso | SIM      |
| Rio Hermawan | PTIK     |
| Rio Hermawan | E-Commerce |
| Rio Hermawan | SIM      |
| Tio Sandewa  | Agama    |
| Tio Sandewa  | Algoritma |
| Agus Supandi | E-Commerce |
| Nurlaela Sari | PTIK     |
| Nurlaela Sari | SIM      |
+-----+-----+
10 rows in set (0.000 sec)
```

Keterangan:

Query ini menampilkan daftar nama dosen beserta nama mata kuliah yang mereka ajarkan dengan menggunakan JOIN antar tabel.

- b. Menampilkan dosen yang mengajar mata kuliah 'E-Commerce'

```
MariaDB [db_join_kampus]> SELECT d.nm_dosen, d.thn_masuk, m.nm_mtkul
-> FROM dosen d
-> JOIN mengajar mg ON d.nip = mg.nip
-> JOIN mt_kuliah m ON mg.kd_mtkul = m.kd_mtkul
-> WHERE m.nm_mtkul = 'E-Commerce';

+-----+-----+-----+
| nm_dosen | thn_masuk | nm_mtkul |
+-----+-----+-----+
| Rio Hermawan | 2009 | E-Commerce |
| Agus Supandi | 2011 | E-Commerce |
+-----+-----+-----+
2 rows in set (0.003 sec)
```

Keterangan:

Menampilkan hanya dosen yang mengajar mata kuliah E-Commerce, disertai tahun masuknya.

- c. Menampilkan daftar mata kuliah dengan jumlah SKS lebih dari 3

```
MariaDB [db_join_kampus]> SELECT kd_mtkul, nm_mtkul, sks
-> FROM mt_kuliah
-> WHERE sks > 3;

+-----+-----+-----+
| kd_mtkul | nm_mtkul | sks |
+-----+-----+-----+
| 101      | Algoritma | 4 |
| 303      | SIM      | 4 |
+-----+-----+-----+
2 rows in set (0.001 sec)
```

Keterangan:

Query ini memfilter data dari tabel mt_kuliah untuk menampilkan hanya mata kuliah dengan beban lebih dari 3 SKS.

- d. Mengurutkan daftar dosen berdasarkan tahun masuk paling awal

```
MariaDB [db_join_kampus]> SELECT nip, nm_dosen, thn_masuk
-> FROM dosen
-> ORDER BY thn_masuk ASC;

+-----+-----+-----+
| nip | nm_dosen | thn_masuk |
+-----+-----+-----+
| 01011 | Budi Santoso | 2003 |
| 01013 | Tio Sandewa | 2005 |
| 01012 | Rio Hermawan | 2009 |
| 01014 | Agus Supandi | 2011 |
| 01015 | Nurlaela Sari | 2014 |
+-----+-----+-----+
5 rows in set (0.001 sec)
```

Keterangan:

Menampilkan data dosen dari tahun masuk terlama hingga terbaru menggunakan ORDER BY ASC.

- e. Menampilkan nama dosen, mata kuliah, dan tahun masuk — urutkan berdasarkan nama mata kuliah (Z-A)

```

MariaDB [db_join_kampus]> SELECT d.nm_dosen AS 'Nama Dosen', m.nm_mtkul AS 'Mengajar', d.thn_masuk
-> FROM dosen d
-> JOIN mengajar mg ON d.nip = mg.nip
-> JOIN mt_kuliah m ON mg.kd_mtkul = m.kd_mtkul
-> ORDER BY m.nm_mtkul DESC;
+-----+-----+-----+
| Nama Dosen | Mengajar | thn_masuk |
+-----+-----+-----+
| Rio Hermawan | SIM      | 2009      |
| Nurlaela Sari | SIM      | 2014      |
| Budi Santoso  | SIM      | 2003      |
| Rio Hermawan | PTIK     | 2009      |
| Nurlaela Sari | PTIK     | 2014      |
| Rio Hermawan | E-Commerce | 2009      |
| Agus Supandi  | E-Commerce | 2011      |
| Tio Sandewa   | Algoritma | 2005      |
| Budi Santoso  | Algoritma | 2003      |
| Tio Sandewa   | Agama     | 2005      |
+-----+-----+-----+
10 rows in set (0.000 sec)

```

Keterangan:

Menampilkan daftar dosen dan mata kuliah yang diajarkan, diurutkan secara descending (Z ke A) berdasarkan nama mata kuliah.