

PERENCANAAN PEMBELAJARAN MENDALAM (PPM)

Identitas	Nama Penyusun : Robie Hakim, S.Kom
	Satuan Pendidikan : SMK Negeri 1 Tanjunganom Nganjuk
	Kelas / Fase : XI / F
	Mata Pelajaran : Konsentrasi Keahlian TKJ
	Prediksi Alokasi Waktu : 10 JP (5 x 45 Menit) 2 Pertemuan
Identifikasi	Peserta Didik: - Sebagian besar telah memahami dasar sistem operasi Linux (Debian/Ubuntu). - Memiliki pengalaman awal dalam virtualisasi sederhana (misalnya VirtualBox), namun belum mengenal Proxmox VE secara mendalam. - Memiliki minat tinggi terhadap teknologi server dan layanan backend. - Membutuhkan pembiasaan disiplin kerja, dokumentasi teknis, dan troubleshooting sistem.
	Materi Pelajaran: - Jenis Pengetahuan: Pengetahuan konseptual (fungsi MariaDB, arsitektur container), prosedural (instalasi, konfigurasi, operasi CRUD), dan metakognitif (troubleshooting & integrasi sistem). - Relevansi: Sangat relevan dengan dunia kerja sebagai teknisi server, administrator jaringan, atau DevOps junior. - Tingkat Kesulitan: Sedang–Tinggi – memerlukan pemahaman sistem operasi, jaringan, dan logika database. - Nilai Karakter: Kedisiplinan, tanggung jawab, problem solving, ketelitian, kerja sama tim.
	Dimensi Profil Lulusan: sesuaikan dengan visi misi sekolah (soft skill) Pilihlah dimensi profil lulusan yang akan dicapai dalam pembelajaran ✓ Kolaborasi ✓ Penalaran Kritis ✓ Kreativitas ✓ Komunikasi
Desain Pembelajaran	Capaian Pembelajaran: Pada akhir Fase F, murid memiliki kemampuan menganalisis konsep database server MariaDB di CT Proxmox Debian 12.
	Lintas Disiplin Ilmu : -

	Tujuan Pembelajaran : 1. Menjelaskan pengertian dan fungsi MariaDB sebagai sistem manajemen basis data relasional dalam lingkungan Container (CT) Proxmox berbasis Debian 12. (C2) 2. Mengimplementasikan instalasi MariaDB pada Container (CT) berbasis Debian 12 di lingkungan Proxmox VE sesuai praktik terbaik sistem operasi dan keamanan. (C3) 3. Memproseskan data melalui operasi CRUD (Create, Read, Update, Delete) pada database MariaDB yang dijalankan di dalam Container Debian 12, serta memverifikasi integritas hasilnya. (C3) 4. Mengatribusikan penyebab umum kegagalan koneksi atau error MariaDB (misalnya: Can't connect to local MySQL server) kepada faktor konfigurasi, jaringan, atau izin sistem di lingkungan Container Debian 12. (C4) 5. Memecahkan masalah integrasi MariaDB dengan aplikasi web (misalnya: PHP + Apache) yang dihosting di Container terpisah namun berada dalam jaringan Proxmox yang sama. (C4)		
	Topik Pembelajaran: 1. Pengantar MariaDB dan arsitektur database server 2. Instalasi dan konfigurasi MariaDB di Container Debian 12 pada Proxmox VE 3. Operasi CRUD dan verifikasi data 4. Troubleshooting error MariaDB umum 5. Integrasi MariaDB dengan aplikasi web di multi-container		
	Praktik Pedagogis: Pendekatan : Pembelajaran Mendalam (Deep Learning) Strategi : Demonstrasi, Praktikum Berbasis Proyek, Problem-Based Learning (PBL) Fokus : Kolaborasi, Pemecahan Masalah, Refleksi Kritis, Dokumentasi Teknis, Pembentukan Karakter Profesional		
	Kemitraan Pembelajaran: -		
	Lingkungan Pembelajaran: Ruang Fisik : Ruang Bengkel Teknik Komputer dan Jaringan Budaya Belajar : Kolaboratif, Reflektif, Berpartisipasi Aktif, Rasa ingin tahu		
	Pemanfaatan Digital:  MediaWiki : https://wiki.robiehakim.my.id/index.php/Install_mariadb_di_Ubuntu_20.04 https://wiki.robiehakim.my.id/index.php/Install_MariaDB_di_Debian_12		

Langkah-Langkah Pembelajaran			
Pengalaman Belajar	Kegiatan Awal		
	1. Guru membuka pembelajaran dengan mengucapkan salam dan meminta salah satu murid memimpin doa 2. Murid berdoa bersama dalam mengawali pembelajaran. 3. Guru mengajak murid untuk menyanyikan lagu Indonesia Raya	15	mnt

	4. Guru memberikan motivasi dan mengecek kehadiran murid ("Sebuah sekolah ingin membuat sistem presensi digital berbasis web. Database-nya harus aman, ringan, dan mudah dikelola. Mengapa MariaDB di Proxmox jadi pilihan?") 5. Guru mengecek kehadiran murid dan memastikan kesiapan laboratorium (akses Proxmox, internet, akun root). 6. Guru bersama murid menyepakati aturan kelas: - Tidak menghapus container milik orang lain - Backup sebelum eksperimen - Gunakan akun non-root untuk operasi database 7. Guru menyampaikan kompetensi dan tujuan pembelajaran hari ini secara eksplisit.		
Kegiatan Inti			
	Pertemuan 1 Tujuan Pembelajaran : - Murid Menjelaskan pengertian dan fungsi MariaDB sebagai sistem manajemen basis data relasional dalam lingkungan Container (CT) Proxmox berbasis Debian 12. - Murid mengimplementasikan instalasi MariaDB pada Container (CT) berbasis Debian 12 di lingkungan Proxmox VE sesuai praktik terbaik sistem operasi dan keamanan. - Murid memproseskan data melalui operasi CRUD (Create, Read, Update, Delete) pada database MariaDB yang dijalankan di dalam Container Debian 12, serta memverifikasi integritas hasilnya. Prinsip: Berkesadaran Bermakna Menggembirakan. Memahami: - Presentasi Konsep (20 menit) - Guru menjelaskan: - Perbedaan MariaDB vs MySQL - Fungsi Sistem Manajemen Basis Data Relasional (RDBMS) - Keunggulan arsitektur Container (CT) vs VM dalam Proxmox - Mengapa Debian 12 dipilih sebagai OS dasar - Gunakan slide digital dan diagram arsitektur (Proxmox → CT → MariaDB) - Demonstrasi Langsung (20 menit) - Guru mendemonstrasikan: - Membuat CT Debian 12 di Proxmox - Masuk ke CLI container - Instalasi MariaDB (<code>apt update && apt install mariadb-server</code>) - Menjalankan <code>mysql_secure_installation</code> - Murid mencatat langkah penting di buku/catatan digital - Diskusi (20 menit) "Apa risiko jika tidak menjalankan <code>mysql_secure_installation</code>?"	200	mnt

2. “Mengapa tidak disarankan login sebagai root dari luar?”

Mengaplikasi: (100 menit)

4. Praktik Kelompok (4–5 orang)

- Setiap kelompok:
 - a. Membuat Container Debian 12 di Proxmox (dengan nama unik, misal: `tkj-mariadb-01`)
 - b. Melakukan instalasi MariaDB
 - c. Menjalankan `mysql_secure_installation` (atur password root, hapus user anonim, nonaktifkan remote root login)
 - d. Login ke MariaDB dan:
 - Membuat database `sekolah`
 - Membuat tabel `siswa` (kolom: `id`, `nama`, `kelas`, `jurusan`)
 - Melakukan operasi CRUD:

```
INSERT INTO siswa VALUES (1, 'Robie', 'XI', 'TKJ');
SELECT * FROM siswa;
UPDATE siswa SET kelas='XII' WHERE id=1;
DELETE FROM siswa WHERE id=1;
```

- e. Memverifikasi hasil dengan perintah `SELECT` dan mencatat output

5. Dokumentasi

- Ambil screenshot setiap tahap:
 - i. Tampilan Proxmox saat membuat CT
 - ii. CLI instalasi MariaDB
 - iii. Hasil operasi CRUD
- Simpan di blogger kelompok (Google Drive/GitHub)

Merefleksi:(40 Menit)

6. Presentasi Singkat (15 menit)

- 2–3 kelompok mempresentasikan:
 - i. Skema database yang dibuat
 - ii. Kendala saat instalasi (misal: error repo, permission denied)
 - iii. Solusi yang diterapkan

7. Penguatan Konsep oleh Guru (15 menit)

- Tekankan:
 - i. Pentingnya user non-root untuk akses aplikasi
 - ii. Prinsip least privilege dalam manajemen database
 - iii. Peran `bind-address` dalam keamanan jaringan

8. Refleksi Tertulis (10 menit)

- Murid menjawab di buku/jurnal digital:

“Apa yang terjadi jika kamu lupa menjalankan

	<code>mysql_secure_installation</code>? Bagaimana itu membahayakan sistem?" Kegiatan Penutup (10 menit) - Guru memandu refleksi lisan: "Apa perbedaan antara <code>CREATE DATABASE</code> dan <code>USE</code>?" "Mengapa operasi DELETE harus hati-hati?" - Murid membersihkan sesi: - Logout dari container - Backup container (opsional, jika memungkinkan) - Guru memberikan preview pertemuan berikutnya: "Besok, kita akan 'sengaja' membuat error MariaDB—lalu memperbaikinya!" - Guru memimpin doa penutup.		
	Pertemuan 2 Tujuan Pembelajaran : - Mengatribusikan penyebab umum kegagalan koneksi atau error MariaDB (misalnya: <i>Can't connect to local MySQL server</i>) kepada faktor konfigurasi, jaringan, atau izin sistem di lingkungan Container Debian 12. - Memecahkan masalah integrasi MariaDB dengan aplikasi web (misalnya: PHP + Apache) yang dihosting di Container terpisah namun berada dalam jaringan Proxmox yang sama. Prinsip: Berkesadaran Bermakna Menggembirakan Memahami - Penjelasan Konsep Troubleshooting (20 menit) - Jenis error umum MariaDB: - Socket error → service tidak jalan - Access denied → user/password salah - Can't connect → <code>bind-address</code>, firewall, atau jaringan - Tools diagnosis: <code>systemctl status mariadb</code> <code>journalctl -u mariadb</code> <code>netstat -tuln grep 3306</code> <code>telnet [IP] 3306</code> - Konsep Jaringan Antar-Container (20 menit) - Proxmox menggunakan bridge (vbr0) - Setiap CT punya IP sendiri di subnet yang sama - MariaDB harus diizinkan menerima koneksi dari IP eksternal (<code>bind-address = 0.0.0.0</code>) - Perlu membuat user dengan hak akses dari host tertentu: <code>CREATE USER 'webuser'@'10.10.10.%' IDENTIFIED BY 'password';</code> <code>GRANT SELECT, INSERT ON sekolah.* TO 'webuser'@'10.10.10.%';</code>	200	mnt

Mengaplikasi (120 menit)

3. Simulasi Error & Diagnosis (60 menit)

- a. Kelompok diminta:
 - a. Matikan service MariaDB → catat error saat akses
 - b. Ubah `bind-address = 127.0.0.1` → coba akses dari luar
 - c. Hapus user → coba koneksi dengan user salah
- b. Gunakan tools diagnosis untuk mengatribusikan penyebab
- c. Perbaiki konfigurasi dan verifikasi

4. Integrasi dengan Aplikasi Web (60 menit)

- a. Buat CT baru dengan Apache + PHP:


```
apt install apache2 php php-mysql
```
- b. Buat file `koneksi.php`:


```
<?php
$host = "IP_MARIADB_CT";
$user = "webuser";
$pass = "password";
$db = "sekolah";
$conn = mysqli_connect($host, $user, $pass, $db);
if ($conn) echo "Koneksi BERHASIL!";
else echo "GAGAL: " .
mysqli_connect_error();
?>
```
- c. Uji akses via browser → jika gagal, diagnosis & perbaiki

Merefleksi (40 menit)

5. Laporan Troubleshooting

- a. Setiap kelompok membuat laporan:
 - i. Gejala error
 - ii. Analisis (tools & log yang digunakan)
 - iii. Solusi
 - iv. Bukti keberhasilan (screenshot)

6. Diskusi Kelas

- a. “Faktor apa yang paling sering menyebabkan gagal koneksi?”
- b. “Mengapa `bind-address = 127.0.0.1` aman tapi tidak untuk multi-container?”

7. Refleksi Tertulis

“Jika kamu jadi admin server, langkah pertama apa yang kamu lakukan saat MariaDB tiba-tiba tidak bisa diakses?”

8.

Kegiatan Penutup			
1. Guru memberikan umpan balik umum.	20	mnt	
2. Murid mengumpulkan dokumentasi akhir (Google Docs/Drive).			
3. Pembersihan peralatan & shutdown container.			
4. Doa penutup.			
.			

Asesmen Pembelajaran	Asesmen pada Awal Pembelajaran:																																	
	Tujuan : Mengukur pemahaman awal siswa tentang sistem operasi Linux, jaringan dasar, virtualisasi, dan konsep database sebelum praktikum. https://docs.google.com/forms/d/e/1FAIpQLSf7-WOFrewB9jdKswq23r_Lol99o5wNcRn79BN1ZwFCibMKQ/viewform?usp=dialog																																	
	Asesmen pada Proses Pembelajaran:																																	
	Rubrik Penilaian Observasi Praktikum																																	
	1. Aspek Teknis (70%)																																	
	<table> <tr> <th>No</th><th>Aspek Penilaian</th><th>Indikator</th><th>Skor 4 (SB)</th><th>Skor 3 (B)</th><th>Skor 2 (C)</th></tr> <tr> <td>1</td><td>Instalasi & Konfigurasi MariaDB</td><td>Instalasi lengkap, "mysql_secure_installation" dijalankan, konfigurasi aman</td><td>Semua langkah benar & aman</td><td>Ada 1 kesalahan kecil</td><td>Beberapa langkah terlewat</td></tr> <tr> <td>2</td><td>Operasi CRUD</td><td>Melakukan semua operasi dengan verifikasi</td><td>CRUD lengkap & terdokumentasi</td><td>CRUD lengkap, kurang verifikasi</td><td>Hanya sebagian operasi</td></tr> <tr> <td>3</td><td>Troubleshooting</td><td>Mampu mendiagnosis & memperbaiki error</td><td>Analisis tepat, solusi efektif</td><td>Analisis benar, solusi kurang optimal</td><td>Analisis kurang tepat</td></tr> <tr> <td>4</td><td>Integrasi Web</td><td>Koneksi PHP–MariaDB berhasil</td><td>Berhasil + dokumentasi jelas</td><td>Berhasil, dokumentasi minim</td><td>Hanya sebagian berhasil</td></tr> </table>					No	Aspek Penilaian	Indikator	Skor 4 (SB)	Skor 3 (B)	Skor 2 (C)	1	Instalasi & Konfigurasi MariaDB	Instalasi lengkap, "mysql_secure_installation" dijalankan, konfigurasi aman	Semua langkah benar & aman	Ada 1 kesalahan kecil	Beberapa langkah terlewat	2	Operasi CRUD	Melakukan semua operasi dengan verifikasi	CRUD lengkap & terdokumentasi	CRUD lengkap, kurang verifikasi	Hanya sebagian operasi	3	Troubleshooting	Mampu mendiagnosis & memperbaiki error	Analisis tepat, solusi efektif	Analisis benar, solusi kurang optimal	Analisis kurang tepat	4	Integrasi Web	Koneksi PHP–MariaDB berhasil	Berhasil + dokumentasi jelas	Berhasil, dokumentasi minim
No	Aspek Penilaian	Indikator	Skor 4 (SB)	Skor 3 (B)	Skor 2 (C)																													
1	Instalasi & Konfigurasi MariaDB	Instalasi lengkap, "mysql_secure_installation" dijalankan, konfigurasi aman	Semua langkah benar & aman	Ada 1 kesalahan kecil	Beberapa langkah terlewat																													
2	Operasi CRUD	Melakukan semua operasi dengan verifikasi	CRUD lengkap & terdokumentasi	CRUD lengkap, kurang verifikasi	Hanya sebagian operasi																													
3	Troubleshooting	Mampu mendiagnosis & memperbaiki error	Analisis tepat, solusi efektif	Analisis benar, solusi kurang optimal	Analisis kurang tepat																													
4	Integrasi Web	Koneksi PHP–MariaDB berhasil	Berhasil + dokumentasi jelas	Berhasil, dokumentasi minim	Hanya sebagian berhasil																													
Bobot Teknis = 70%																																		

2. Aspek Sikap (30%)

No	Indikator Penilaian	Skor 4 (Sangat Baik)	Skor 3 (Baik)	Skor 2 (Cukup)	Skor 1 (Kurang)
6	Kerja Sama	Kontribusi aktif dalam kelompok	Sangat aktif & membantu	Aktif	Pasif
7	Disiplin & Tanggung Jawab	Tepat waktu, ikuti instruksi	Selalu disiplin	Umumnya disiplin	Kadang lalai
8	Keselamatan Kerja	Patuhi aturan keamanan	Selalu patuh, rapi	Umumnya patuh	Kadang lalai

Bobot Sikap = 30%

Asesmen pada Akhir Pembelajaran

Model : Uji Kinerja (Performance Test)

Topik : Instalasi, Operasi CRUD, dan Integrasi MariaDB di Container Proxmox Debian 12

Instruksi Kerja

Anda bertindak sebagai administrator server di sebuah perusahaan IT. Tugas Anda adalah:

1. Menginstal MariaDB di dalam Container (CT) Debian 12 pada lingkungan Proxmox VE.
2. Membuat database **sekolah** dan tabel **siswa**, lalu melakukan operasi CRUD (Create, Read, Update, Delete).
3. Mengonfigurasi MariaDB agar dapat diakses dari Container lain (misalnya untuk aplikasi web).
4. Membuat Container kedua berisi Apache + PHP, lalu menguji koneksi ke MariaDB menggunakan script PHP.
5. Jika terjadi error koneksi, lakukan diagnosis dan perbaikan.
6. Menyusun laporan teknis singkat yang mencakup:
 - Langkah-langkah pengerjaan
 - Hasil operasi CRUD
 - Bukti koneksi berhasil (screenshot output PHP)
 - Analisis jika terjadi error dan solusi yang diterapkan

Alat & Bahan

- Server Proxmox VE (versi ≥ 7.0) dengan akses root
- 2 buah Container (CT) berbasis Debian 12
- Koneksi internet pada Container
- Terminal/SSH client (PuTTY, Terminal Linux, dsb)
- Browser (untuk uji aplikasi web)

Langkah Kerja

1. Persiapan
 - Buat Container Debian 12 untuk MariaDB (`mariadb-<nama>`)
 - Buat Container Debian 12 untuk Web (`web-<nama>`)
2. Instalasi & Konfigurasi MariaDB
 - Instal MariaDB di CT MariaDB
 - Jalankan `mysql_secure_installation`
 - Ubah `bind-address = 0.0.0.0` pada file konfigurasi
 - Restart service MariaDB
3. Operasi Database
 - Buat database `sekolah` dan tabel `siswa`
 - Lakukan operasi CRUD lengkap
 - Buat user `webuser` dengan hak akses dari subnet jaringan Proxmox
4. Integrasi Web
 - Instal Apache + PHP di CT Web
 - Buat file `koneksi.php` untuk menguji koneksi ke MariaDB
 - Akses file tersebut via browser
5. Troubleshooting (Simulasi Error)
 - Matikan service MariaDB → catat error
 - Perbaiki dan verifikasi koneksi kembali
6. Dokumentasi
 - Ambil screenshot setiap tahap kritis
 - Susun laporan teknis dalam format digital (Google Docs/PDF)

Indikator Penilaian

1. **Persiapan:** Kesiapan Container, jaringan, dan keamanan kerja
2. **Proses:** Ketepatan prosedur instalasi, konfigurasi, dan integrasi
3. **Hasil:** Keberhasilan operasi CRUD dan koneksi antar-container
4. **Pemecahan Masalah:** Kemampuan mendiagnosis dan memperbaiki error
5. **Sikap Kerja:** Disiplin, kerja sama, ketelitian, dan keselamatan kerja
6. **Dokumentasi:** Kelengkapan, kejelasan, dan kerapian laporan teknis

Rubrik Penilaian

Aspek	Indikator	Skor 1 (Kurang)	Skor 2 (Cukup)	Skor 3 (Baik)	Skor 4 (Sangat Baik)
Persiapan	Kesiapan alat & Container	Tidak lengkap, tidak siap digunakan	Lengkap, tapi konfigurasi awal salah	Lengkap, siap digunakan	Lengkap, rapi, dan siap optimal
Proses	Instalasi & konfigurasi MariaDB	Banyak kesalahan prosedur	Beberapa langkah terlewat	Hampir sempurna, minor error	Benar & sesuai best practice
Hasil	Operasi CRUD & koneksi web	Gagal total	Hanya sebagian berhasil	Berhasil, tapi kurang verifikasi	Berhasil 100% + terdokumentasi
Pemecahan Masalah	Diagnosis & perbaikan error	Tidak mampu mendiagnosis	Analisis kurang tepat	Solusi benar, lambat	Cepat, akurat, sistematis
Sikap Kerja	Disiplin, kerja sama, K3	Tidak disiplin, abai K3	Kadang lalai	Umumnya baik	Sangat profesional

Skor Akhir

- Teknis (Persiapan + Proses + Hasil + Dokumentasi) = 70%
- Sikap (Kerjasama, Disiplin, Tanggung jawab) = 30%

Nilai Akhir = $(\text{Total Skor} \div 20) \times 100$

Nganjuk, 14 Juli 2025

Kepala
SMK Negeri 1
Tanjunganom

Waka. Kurikulum

Guru

Harbudi Susilo, M.Pd
NIP.19770704200801 1010

Mohammad Najmudin, S.Kom, M.Pd
NIP.198201122009031004

Robie Hakim, S.Kom
NIP.198808202022211013

LAMPIRAN

JOB SHEET PRAKTIKUM

Mata Pelajaran : Konsentrasi Keahlian Teknik Komputer dan Jaringan (TKJ)

Kelas / Semester : XI / Ganjil

Topik : Instalasi, Operasi CRUD, dan Integrasi MariaDB di Container Proxmox Debian 12

Alokasi Waktu : 2 × 225 menit (2 pertemuan)

Tujuan Praktikum

1. Siswa mampu menjelaskan fungsi dan arsitektur MariaDB dalam lingkungan container Proxmox.
2. Siswa mampu menginstal dan mengonfigurasi MariaDB di Container Debian 12 sesuai praktik keamanan terbaik.
3. Siswa mampu melakukan operasi CRUD dan memverifikasi integritas data.
4. Siswa mampu mendiagnosis error umum MariaDB dan memperbaikinya.
5. Siswa mampu mengintegrasikan MariaDB dengan aplikasi web (PHP + Apache) di container terpisah.

Dasar Teori Singkat

MariaDB adalah sistem manajemen basis data relasional (RDBMS) open-source yang kompatibel dengan MySQL. Dalam lingkungan virtualisasi seperti Proxmox VE, MariaDB dapat dijalankan di dalam Container (CT) berbasis LXC untuk efisiensi sumber daya. Integrasi dengan aplikasi web memerlukan konfigurasi jaringan antar-container, manajemen user database, dan pengujian konektivitas.

Alat dan Bahan

- Server Proxmox VE (versi ≥ 7.0)
- Akses root ke Proxmox
- 2 buah Container (CT) berbasis Debian 12
 - CT-1: Untuk MariaDB
 - CT-2: Untuk Apache + PHP
- Koneksi internet pada CT
- Terminal/SSH client (PuTTY, Terminal Linux, dsb)
- Browser (untuk uji aplikasi web)

Keselamatan Kerja

- Jangan menghapus container milik kelompok lain.
- Backup container sebelum eksperimen berisiko.
- Gunakan akun non-root untuk akses database aplikasi.
- Jangan membagikan password root MariaDB.

Langkah Kerja

Pertemuan 1: Instalasi & Operasi CRUD

1. Persiapan

- Buat CT Debian 12 di Proxmox (nama: `mariadb-<no-kelompok>`)
- Masuk ke CLI container via Proxmox Console atau SSH

2. Instalasi MariaDB

```
apt update && apt install mariadb-server -y  
mysql_secure_installation
```

- Ikuti petunjuk: atur password root, hapus user anonim, nonaktifkan remote root login

3. Operasi CRUD

- Login ke MariaDB: `mysql -u root -p`
- Jalankan perintah SQL berikut:

```
CREATE DATABASE sekolah;  
  
USE sekolah;  
  
CREATE TABLE siswa (id INT PRIMARY KEY, nama VARCHAR(50), kelas  
VARCHAR(10));  
  
INSERT INTO siswa VALUES (1, 'Robie', 'XI');  
  
SELECT * FROM siswa;  
  
UPDATE siswa SET kelas='XII' WHERE id=1;  
  
DELETE FROM siswa WHERE id=1;
```

4. Dokumentasi

- Ambil screenshot setiap output perintah
- Simpan dalam folder kelompok

Pertemuan 2: Troubleshooting & Integrasi Web

1. Buat CT Aplikasi Web

- Buat CT baru (nama: `web-<no-kelompok>`)
- Instal Apache + PHP:

```
apt install apache2 php php-mysql -y
```

2. Konfigurasi MariaDB untuk Akses Eksternal

- Di CT MariaDB, edit:

```
nano /etc/mysql/mariadb.conf.d/50-server.cnf
```

- Ubah baris:

```
bind-address = 0.0.0.0
```

Restart MariaDB: `systemctl restart mariadb`

3. Buat User Database untuk Web

```
CREATE USER 'webuser'@'10.10.10.%' IDENTIFIED BY 'password123';  
GRANT SELECT, INSERT, UPDATE, DELETE ON sekolah.* TO  
'webuser'@'10.10.10.%;  
FLUSH PRIVILEGES;
```

4. Uji Koneksi via PHP

- Buat file di CT Web: `/var/www/html/koneksi.php`

```
<?php  
$conn = mysqli_connect("IP_MARIADB_CT", "webuser",  
"password123", "sekolah");  
if ($conn) echo "✅ Koneksi BERHASIL!";  
else echo "❌ GAGAL: " . mysqli_connect_error();  
?>
```

- Akses via browser: `http://IP_WEB_CT/koneksi.php`

5. Simulasi & Perbaikan Error

- Matikan service MariaDB → catat error
- Ubah `bind-address` kembali ke `127.0.0.1` → uji koneksi
- Gunakan `systemctl`, `journalctl`, `netstat`, `telnet` untuk diagnosis
- Perbaiki dan verifikasi

6. Dokumentasi Akhir

- Laporkan: gejala → analisis → solusi → bukti keberhasilan

Pertanyaan Refleksi

- Mengapa `bind-address = 0.0.0.0` diperlukan untuk akses antar-container?
- Apa risiko keamanan jika menggunakan user root untuk koneksi aplikasi?
- Bagaimana cara memastikan data tidak hilang saat container dihapus?
- Jika `telnet IP_MARIADB 3306` gagal, apa kemungkinan penyebabnya?

2. RUBRIK PENILAIAN TERPERINCI

NO	Aspek Penilaian	Indikator	Skor 4	Skor 3	Skor 2	Skor 1
TEKNIS (70%)						
1	Instalasi & Konfigurasi MariaDB	Instalasi lengkap, <code>mysql_secure_installation</code> dijalankan, konfigurasi aman	Semua langkah benar & aman	Ada 1 kesalahan kecil	Beberapa langkah terlewat	Tidak berhasil instal
2	Operasi CRUD	Melakukan semua operasi dengan verifikasi	CRUD lengkap & terdokumentasi	CRUD lengkap, kurang verifikasi	Hanya sebagian operasi	Tidak melakukan CRUD
3	Troubleshooting	Mampu mendiagnosis & memperbaiki error	Analisis tepat, solusi efektif	Analisis benar, solusi kurang optimal	Analisis kurang tepat	Tidak mampu diagnosis

4	Integrasi Web	Koneksi PHP–MariaDB berhasil	Berhasil + dokumentasi jelas	Berhasil, dokumen tasi minim	Hanya sebagian berhasil	Gagal total
5	Dokumenta si	Laporan lengkap (screenshot, log, analisis)	Sistematis, rapi, lengkap	Lengkap tapi kurang rapi	Sebagian ada	Tidak ada

SIKAP (30%)

6	Kerja Sama	Kontribusi aktif dalam kelompok	Sangat aktif & membantu	Aktif	Pasif	Tidak koopera tif
7	Disiplin & Tanggung Jawab	Tepat waktu, ikuti instruksi	Selalu disiplin	Umumnya disiplin	Kadang lalai	Tidak bertanggung jawab
8	Keselamatan Kerja	Patuhi aturan keamanan	Selalu patuh, rapi	Umumnya patuh	Kadang lalai	Abaikan aturan

Skor Akhir = $(\sum \text{Skor Teknis} \times 0.7) + (\sum \text{Skor Sikap} \times 0.3)$

Nilai = $(\text{Skor Akhir} / 32) \times 100$

Lembar Hasil Praktikum

No	Kegiatan Praktikum	Berhasil / Tidak	Catatan / Kendala
1	Membuat Container Debian 12 di Proxmox VE	☐ Berhasil ☐ Tidak	
2	Melakukan instalasi MariaDB di Container	☐ Berhasil ☐ Tidak	

3	Menjalankan mysql_secure_installation dan mengatur keamanan awal	☐ Berhasil ☐ Tidak	
4	Membuat database sekolah dan tabel siswa	☐ Berhasil ☐ Tidak	
5	Melakukan operasi CRUD (Create, Read, Update, Delete)	☐ Berhasil ☐ Tidak	
6	Memverifikasi integritas data setelah operasi CRUD	☐ Berhasil ☐ Tidak	
7	Mengonfigurasi bind-address = 0.0.0.0 pada MariaDB	☐ Berhasil ☐ Tidak	
8	Membuat user database khusus untuk aplikasi web (webuser)	☐ Berhasil ☐ Tidak	
9	Membuat Container baru untuk Apache + PHP	☐ Berhasil ☐ Tidak	
10	Membuat file koneksi.php dan menguji koneksi ke MariaDB	☐ Berhasil ☐ Tidak	
11	Mensimulasikan error MariaDB (misal: matikan service, ubah konfigurasi)	☐ Berhasil ☐ Tidak	
12	Mendiagnosis penyebab error menggunakan systemctl"journalctl netstat atau telnet"	☐ Berhasil ☐ Tidak	
13	Memperbaiki konfigurasi dan memulihkan koneksi	☐ Berhasil ☐ Tidak	
14	Mendokumentasikan seluruh proses (screenshot, log, analisis)	☐ Berhasil ☐ Tidak	

Pertanyaan Refleksi

1. Mengapa konfigurasi `bind-address = 0.0.0.0` diperlukan agar MariaDB dapat diakses dari Container lain, dan apa risiko keamanannya jika tidak dikendalikan dengan benar?
2. Apa yang terjadi jika Anda menggunakan akun `root` MariaDB untuk koneksi aplikasi web? Jelaskan dampaknya terhadap keamanan sistem!
3. Saat terjadi error "Can't connect to MySQL server", langkah-langkah diagnosis apa saja yang Anda lakukan, dan tool apa yang paling membantu dalam mengidentifikasi akar masalah?
4. Bagaimana prinsip *least privilege* (hak akses minimal) dapat diterapkan dalam manajemen user MariaDB untuk mendukung praktik administrasi server yang aman dan profesional?

Penilaian

- Teknis (70%): Kebenaran pemasangan, kerapian, hasil uji konektivitas.
- Sikap (30%): Kerjasama, disiplin, tanggung jawab, keselamatan kerja.