

TUGAS 11 – MANAJEMEN KELAS DAN MAHASISWA BERBASIS WEB
PELATIHAN JAVA FUNDAMENTAL KELOMPOK 2

“Aplikasi Manajemen Kelas dan Mahasiswa (One-to-Many Mapping)”

Silahkan buat sebuah **Aplikasi Web Manajemen Kelas dan Mahasiswa** menggunakan:

- Java Spring Boot
- Thymeleaf
- Spring MVC
- Spring Data JPA
- MySQL Database
- Association Mapping (One-to-Many & Many-to-One)

Project mencakup fitur:

- CRUD data Kelas
- CRUD data Mahasiswa
- Mahasiswa harus terkait dengan sebuah Kelas (relasi Many-to-One)
- Satu Kelas boleh memiliki banyak Mahasiswa (relasi One-to-Many)

Aplikasi harus menampilkan hubungan tersebut di UI (Thymeleaf).

Spesifikasi Project

1. Database Structure

Buat database:

db_class_management

Minimal memiliki 2 tabel:

TABLE 1 - classes

Isi kolomnya => id, name, description, created_at

TABLE 2 - students

Isi kolomnya => id, name, email, class_id, created_at

2. Fitur CRUD

A. Fitur Kelas

- Tambah kelas
- Lihat semua kelas
- Edit kelas
- Hapus kelas
- Tampilkan jumlah mahasiswa dalam tiap kelas (opsional)

B. Fitur Mahasiswa

- Tambah mahasiswa ke kelas tertentu
-> Form harus punya dropdown list berisi semua kelas
- Lihat semua mahasiswa
- Edit data mahasiswa termasuk pindah kelas
- Hapus mahasiswa
- Tampilkan mahasiswa yang termasuk dalam suatu kelas

3. Halaman Thymeleaf Wajib

1. classes.html

Tabel berisi semua kelas:

No	Nama Kelas	Total Mahasiswa	Aksi

2. class-detail.html

Menampilkan detail kelas + daftar mahasiswa di kelas tersebut.

3. students.html

Tabel berisi seluruh mahasiswa + informasi nama kelas.

4. add-class.html & edit-class.html

5. add-student.html & edit-student.html

Form harus menampilkan **dropdown kelas**.

4. Syarat Teknis

- Wajib Spring Data JPA Repository
- Tidak boleh pakai JDBC Template (fokus relasi)
- Validasi minimal:
 - Nama tidak boleh kosong

- Email harus valid dan unique
- Mahasiswa wajib memilih kelas
- Cascading optional (boleh diaktifkan atau tidak)

5. Routing Minimal

Kelas

Route	Method	Deskripsi
/classes	GET	List kelas
/classes/add	GET	Form tambah kelas
/classes/add	POST	Simpan kelas
/classes/edit/{id}	GET	Form edit
/classes/edit	POST	Update
/classes/delete/{id}	GET/POST	Delete
/classes/{id}	GET	Detail kelas + mahasiswa

Mahasiswa

Route	Method	Deskripsi
/students	GET	List mahasiswa
/students/add	GET	Form tambah
/students/add	POST	Simpan
/students/edit/{id}	GET	Form edit
/students/edit	POST	Update
/students/delete/{id}	GET/POST	Hapus

6. Output Akhir yang Wajib Dikumpulkan

1. Link repo proyek
2. File SQL tabel (classes & students)
3. Screenshot halaman:
 - Daftar kelas

- Daftar mahasiswa
- Form tambah mahasiswa
- Halaman detail kelas

4. **README.md** berisi:

- Penjelasan fitur
- Penjelasan relasi entity
- Struktur folder

5. **LINK YouTube** (harus ada wajah kalian saat sedang menjelaskan, bisa disimpan dalam video di bagian pojok kanan atau kiri bawah berukuran kecil), berisi:

- Demo program
- Penjelasan alur Controller → Service → Repository
- Penjelasan One-to-Many dan Many-to-One
- Durasi minimal 3–5 menit