

Nama : Kurnia Feby Vidayanto

NIM : L200170102

Kelas : D

Modul : 2

Nomor 1

1. Menentukan entities (object-object dasar) yang perlu ada di database.

- **mahasiswa** : menyimpan semua data pribadi semua mahasiswa
- **dosen** : menyimpan semua data pribadi semua dosen
- **mata_kuliah** : menyimpan semua data mata kuliah
- **ruang_kelas** : menyimpan informasi tentang semua ruang kelas

2. Menentukan attributes (sifat-sifat) masing-masing entity sesuai kebutuhan database

- **mahasiswa :**
 - ✓ NIM : nomor identitas untuk mahasiswa (integer) PK
 - ✓ nama : nama untuk semua mahasiswa (varchar(45))
 - ✓ alamat : alamat lengkap mahasiswa (varchar(255))
- **dosen :**
 - ✓ NIP : nomor identitas untuk dosen (integer) PK
 - ✓ nama : nama lengkap untuk dosen (varchar(45))
 - ✓ alamat : alamat lengkap dosen (varchar(255))
- **mata_kuliah :**
 - ✓ kode_mk : kode untuk mata kuliah (varchar(10)) PK
 - ✓ nama_mk : nama untuk mata kuliah (varchar(30))
- **ruang_kelas :**
 - ✓ kode_ruang : kode untuk ruang kelas (varchar(10)) PK
 - ✓ nama_ruang : nama untuk ruang kelas (varchar(30))

3. Menentukan relationship (hubungan) antar entitas

	mahasiswa	dosen	mata_kuliah	ruang_kelas
mahasiswa	-	m:n	m:n	-
dosen		-	m:n	m:n
mata_kuliah			-	m:n
ruang_kelas				-

Hubungan



dosen mengajar mahasiswa

- ✓ Tabel utama : dosen, mahasiswa
- ✓ Tabel kedua : dosen_teach_mahasiswa
- ✓ Relationship : many-to-many (m:n)
- ✓ Attribute penghubung : NIM, NIP (FK NIM, NIP di dosen_teach_mahasiswa)



mahasiswa mengambil mata_kuliah

- ✓ Tabel utama : mahasiswa, mata_kuliah
- ✓ Tabel kedua : mahasiswa_take_mata_kuliah
- ✓ Relationship : many-to-many (m:n)
- ✓ Attribute penghubung : NIM, kode_mk (FK NIM, kode_mk di mahasiswa_take_mata_kuliah)



dosen mengampu mata_kuliah

- ✓ Tabel utama : dosen, mata_kuliah
- ✓ Tabel kedua : dosen_keepup_mata_kuliah
- ✓ Relationship : many-to-many (m:n)
- ✓ Attribute penghubung : NIP, kode_mk (FK NIP, kode_mk di dosen_keepup_mata_kuliah)



dosen menggunakan ruang_kelas

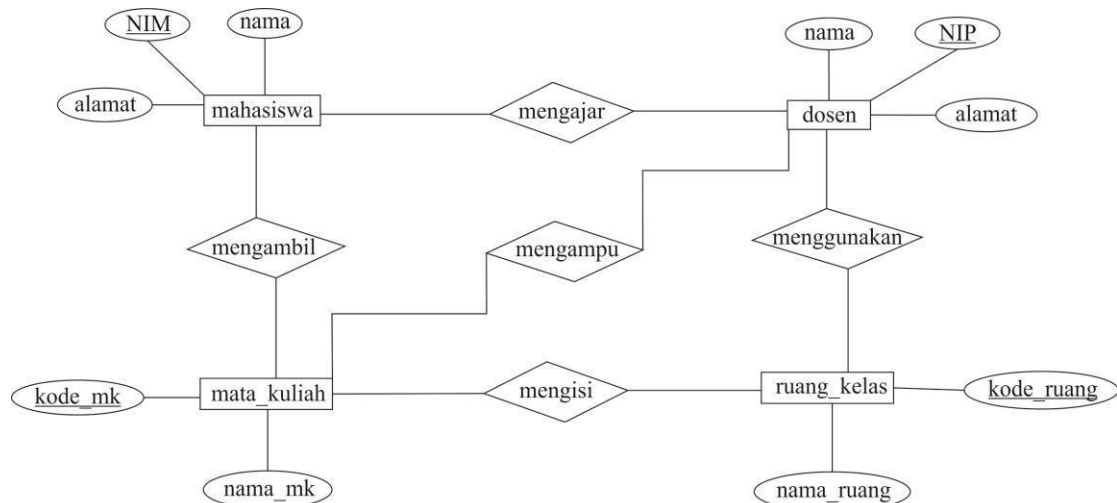
- ✓ Tabel utama : dosen, ruang_kelas

- ✓ Tabel kedua : dosen_use_ruang_kelas
- ✓ Relationship : many-to-many (m:n)
- ✓ Attribute penghubung : NIP, kode_ruang (FK NIP, kode_ruang di dosen_use_ruang_kelas)

● **mata_kuliah mengisi ruang_kelas**

- ✓ Tabel utama : mata_kuliah, ruang_kelas
- ✓ Tabel kedua : mata_kuliah_fill_ruang_kelas
- ✓ Relationship : many-to-many (m:n)
- ✓ Attribute penghubung : kode_mk, kode_ruang (FK kode_mk, kode_ruang di mata_kuliah_fill_ruang_kelas)

4. Menggambar ER Diagram



Nomor 2

1. Menentukan entities (object-object dasar) yang perlu ada di database.

- **anggota** : menyimpan semua data pribadi semua anggota perpustakaan
- **pegawai** : menyimpan semua data pribadi semua pegawai perpustakaan
- **buku** : menyimpan semua data buku di perpustakaan
- **denda** : menyimpan informasi tentang denda

2. Menentukan attributes (sifat-sifat) masing-masing entity sesuai kebutuhan database

- **anggota** :
 - ✓ no_anggota : nomor identitas untuk anggota perpustakaan (integer) PK
 - ✓ nama : nama untuk semua anggota perpustakaan (varchar(45))
 - ✓ alamat : alamat lengkap anggota perpustakaan (varchar(255))
 - ✓ tgl_lahir : tanggal lahir semua anggota perpustakaan (varchar(45))
 - ✓ jurusan : jurusan semua anggota perpustakaan (varchar(45))
- **pegawai** :
 - ✓ no_pegawai : nomor identitas untuk pegawai perpustakaan (integer) PK
 - ✓ nama : nama untuk semua pegawai perpustakaan (varchar(45))
 - ✓ alamat : alamat lengkap pegawai perpustakaan (varchar(255))
 - ✓ no_tlp : nomor telp semua pegawai perpustakaan (integer)
 - ✓ jabatan : jabatan dari semua pegawai perpustakaan (varchar(45))
- **buku** :
 - ✓ no_buku : nomor untuk buku perpustakaan (integer) PK
 - ✓ judul : judul dari buku perpustakaan (varchar(45))
 - ✓ pengarang : pengarang dari buku perpustakaan (varchar(45))
 - ✓ thn_terbit : tahun terbit dari buku perpustakaan (integer)
 - ✓ penerbit : penerbit dari buku perpustakaan (varchar(45))
- **denda** :
 - ✓ kode_denda : kode denda dari perpustakaan (integer) PK
 - ✓ tarif_denda : tarif dari denda di perpustakaan (varchar(45))
 - ✓ jenis_denda : jenis denda di perpustakaan (varchar(45))

- ✓
tgl_pinjam : tanggal peminjaman buku di perpustakaan (varchar(45))
- ✓
no_anggota : nomor identitas untuk anggota perpustakaan (integer)

3. Menentukan relationship (hubungan) antar entitas

	anggota	pegawai	buku	denda
anggota	-	n:1	m:n	1:n
pegawai		-	m:n	-
buku			-	-
denda				-

● Hubungan

● anggota meminjam buku

- ✓ Tabel utama : anggota, buku
- ✓ Tabel kedua : anggota_borrow_buku
- ✓ Relationship : many-to-many (m:n)
- ✓ Attribute penghubung : no_anggota, no_buku (FK no_anggota, no_buku di anggota_borrow_buku)

● anggota bayar denda

- ✓ Tabel utama : anggota
- ✓ Tabel kedua : denda
- ✓ Relationship : one-to-many (1:n)
- ✓ Attribute penghubung : no_anggota (FK no_anggota di denda)

● pegawai melayani anggota

- ✓ Tabel utama : pegawai
- ✓ Tabel kedua : anggota
- ✓ Relationship : one-to-many (1:n)
- ✓ Attribute penghubung : no_pegawai (FK no_pegawai di anggota)

● pegawai mendata buku

- ✓ Tabel utama : pegawai, buku

- ✓ Tabel kedua : pegawai_register_buku
- ✓ Relationship : many-to-many (m:n)
- ✓ Attribute penghubung : no_pegawai, no_buku (FK no_pegawai, no_buku di pegawai_register_buku)

4. Menggambar ER Diagram

