

**Nama : DANANG AJI N**

**NIM : L200180015**

**Kelas : A**

Langkah-langkah perancangan database data-data kuliah. Menentukan entitas (object-object dasar) yang perlu ada di database.

- a. **Mahasiswa** menyimpan semua data pribadi semua mahasiswa
  - b. **Dosen** : menyimpan semua data pribadi semua dosen
  - c. **Mata kuliah** : menyimpan informasi tentang semua mata kuliah yang ada
  - d. **Ruang kelas** : menyimpan informasi tentang semua ruang kelas
- Menentukan attributes (sifat-sifat) masing-masing entity sesuai kebutuhan database

**a. Mahasiswa:**

- id\_mahasiswa: nomor id untuk mahasiswa (integer) PK
- nama\_mahasiswa: nama lengkap mahasiswa (varchar(45))
- alamat\_mahasiswa: alamat lengkap nasabah (varchar(255))

**b. Dosen:**

- id\_dosen: nomor id untuk dosen (integer) PK
- nama\_dosen: nama lengkap mahasiswa (varchar(45))
- kode\_dosen: kode untuk dosen (varchar(50)) PK
- alamat\_dosen: alamat lengkap nasabah (varchar(255))

**c. Mata kuliah:**

- nama\_mata\_kuliah: nama mata kuliah

- kode\_mata\_kuliah: kode untuk mata kuliah (sistem basis data(50))

**d. Ruang kelas:**

- Jumlah: jumlah kelas dalam universitas
- nama\_ruang\_kelas: nama ruang kelas

- Menentukan relationship (hubungan) antar entitas

	<b>mahasiswa</b>	<b>Dosen</b>	<b>mata_kuliah</b>	<b>ruang_kelas</b>
<b>mahasiswa</b>	-	<b>m : n</b>	<b>m : n</b>	-
<b>dosen</b>		-	<b>1 : n</b>	-
<b>mata_kuliah</b>			-	<b>1 : 1</b>
<b>ruang_kelas</b>	-	-		-

**Hubungan**

**a. mahasiswa diampu dosen:**

- Tabel utama: **mahasiswa, dosen**
- Tabel kedua: mahasiswa\_diampu\_dosen
- Relationship: many-to-one (n:1)
- Attribute penghubung: **id\_mahasiswa, kode\_dosen** (FK id\_mahasiswa, kode\_dosen di mahasiswa\_diampu\_dosen)

**b. mahasiswa mengambil mata\_kuliah:**

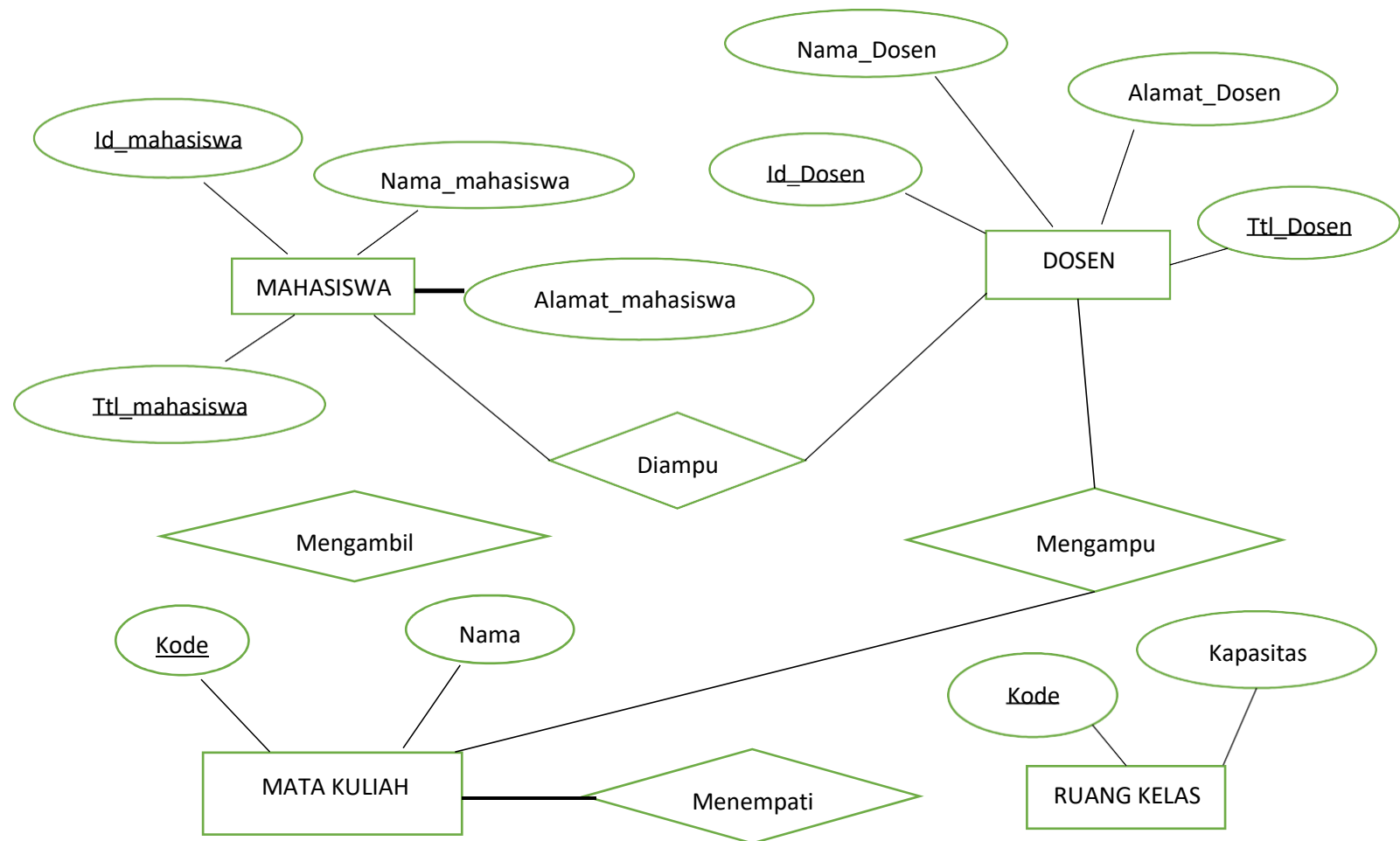
- Tabel utama: **mahasiswa**
- Tabel kedua: **mata\_kuliah**
- Relationship: one-to-many (1:n)
- Attribute penghubung: **id\_mahasiswa, kode\_mata\_kuliah** (FK id\_mahasiswa, kode\_mata\_kuliah di mahasiswa\_mengambil\_mata\_kuliah)

**c. dosen mengampu mata\_kuliah:**

- Tabel utama: **dosen**
- Tabel kedua: **mata\_kuliah**
- Relationship: one-to-many (1:n)
- Attribute penghubung: **id\_dosen, kode\_mata\_kuliah** (FK id\_dosen, kode\_mata\_kuliah di dosen\_mengampu\_mata\_kuliah)

**d. Mata kuliah menempati ruang\_kelas:**

- Tabel utama : **mata\_kuliah**
- Tabel kedua : **ruang\_kelas**
- Relationship : One-to-one (1:1)
- Attribute penghubung : **Kode\_matakuliah, Kode\_ruangkelas**
- Menggambar ERD Diagram



2. Langkah-langkah perancangan database data-data kuliah.

- Menentukan entitas (object-object dasar) yang perlu ada di database.
  - a. **Anggota:** menyimpan semua data pribadi semua anggota perpustakaan
  - b. **Pegawai:** menyimpan semua data pribadi semua pegawai perpustakaan
  - c. **Buku:** menyimpan semua data buku di perpustakaan
  - d. **Denda:** menyimpan informasi tentang denda
- Menentukan attributes (sifat-sifat) masing-masing entity sesuai kebutuhan database
  - a. **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))
  - b. **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))
  - c. **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))
  - d. **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)

- Menentukan relationship (hubungan) antar entitas

	<b>anggota</b>	<b>pegawai</b>	<b>denda</b>	<b>buku</b>
<b>anggota</b>	-	<b>n : 1</b>	<b>m : n</b>	<b>1 : n</b>
<b>pegawai</b>		-	<b>m : n</b>	-
<b>denda</b>			-	-
<b>buku</b>				-

### Hubungan

**a. anggota meminjam buku:**

- Tabel utama: **anggota, buku**
- Tabel kedua: **anggota\_meminjam\_buku**
- Relationship: many-to-many (m:n)
- Attribute penghubung: **no\_anggota, no\_buku** (FK no\_anggota, no\_buku di **anggota\_borrow\_buku**)

**b. anggota membayar denda:**

- Tabel utama: **anggota**
- Tabel kedua: **denda**
- Relationship: one-to-many (1:n)
- Attribute penghubung: **no\_anggota** (FK no\_anggota di denda)

**c. pegawai melayani anggota:**

- Tabel utama: **pegawai**
- Tabel kedua: **anggota**
- Relationship: one-to-many (1:n)
- Attribute penghubung: **no\_pegawai** (FK no\_pegawai di anggota)

**d. pegawai mendata buku:**

- Tabel utama: **pegawai**
- Tabel kedua: **buku**
- Relationship: Many-to-many (m:n)
- Attribute penghubung: **no\_pegawai, no\_buku** (FK no\_pegawai, no\_buku di **pegawai\_register\_buku**)

- Menggambar ERD Diagram

