Nama : Palistya Nesty H

NIM : L200170083

Kelas : D

Tugas!

1. Rancangan database mengenai mahasiswa, dosen, ruang kelas, dan mata kuliah!

Langkah-langkah perancangan database:

1. Menentukan entities:
2. Mahasiswa: menyimpan semua data pribadi semua mahasiswa
3. Dosen : menyimpan semua data pribadi semua dosen
4. Mata\_kuliah : menyimpan semua data mata kuliah
5. Ruang\_kelas : menyimpan informasi semua ruang kelas
6. Menentukan atribute(sifat-sifat) masing-masing entiti sesuai kebutuhan database
7. Mahasiswa

* Nama\_mahasiswa: nama lengkap mahasiswa (varchar(45))
* NIM\_mahasiswa: nomor induk mahasiswa (varchar(25)) PK
* Alamat\_mahasiswa: alamat lengkap mahasiswa (varchar(255))

1. Dosen

* Nama\_dosen: nama lengkap dosen (varchar(45))
* NIP\_dosen: nomor induk pegawai dosen (varchar(24)) PK
* Alamat: alamat lengkap dosen (varchar(255))

1. Mata\_kuliah

* Nama\_mata kuliah: nama mata kuliah yang diajarkan(varchar(25))
* Kode\_MK: kode setiap mata kuliah (varchar(25)) PK

1. Ruang\_kelas

* Nama\_ruang: nama ruang kelas(varchar(25))
* Kode\_ruang: kode ruang kelas(varchar(25)) PK

1. Menentukan relationship (hubungan) antar entitas

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | mahasiswa | dosen | Mata\_kuliah | Ruang\_kelas |
| mahasiswa | - | m:n | m:n | n:1 |
| dosen |  | - | m:n | n:1 |
| Mata\_kuliah |  |  | - | - |
| Ruang\_kelas |  |  |  | - |

Hubungan

* Mahasiswa diampu dosen:
* Tabel utama: mahasiswa, dosen
* Tabel kedua: mahasiswa\_has\_dosen
* Relationship: many-to-many (m:n)
* Atribute pengubung: NIM\_mahasiswa, NIP\_dosen, (FK NIM, NIP di Mahasiswa  
  \_has\_Dosen)
* Mahasiswa mengambil Mata\_kuliah
* Tabel utama: mahasiswa, Mata\_kuliah
* Tabel kedua: mahasiswa\_has\_mata\_kuliah
* Relationship: many-to-many(m:n)
* Atribute penghubung: nama\_mahasiswa, kode\_MK (FK nama\_mahasiswa, kode\_MK di mahasiswa\_has\_mata\_kuliah)
* Mahasiswa menempati Ruang\_kelas
* Tabel utama: mahasiswa, ruang\_kelas
* Tabel kedua: ruang\_kelas
* Relationship: one-to-many(1:n)
* Atribute penghubung: nim\_mahasiswa (FK nim\_mahasiswa di ruang\_kelas)
* Dosen mengampu Mata\_kuliah
* Tabel utama: dosen, mata\_kuliah
* Tabel kedua: dosen\_has\_mata\_kuliah
* Relationship: many-to-many(m:n)
* Atribute penghubung: NIP\_dosen, kode\_MK (FK NIP\_dosen, kode\_MK di dosen\_has\_mata\_kuliah)
* Dosen menempati ruang\_kelas
* Tabel utama: dosen, ruang\_kelas
* Tabel kedua: ruang\_kelas
* Relationship: one-to-many(1:n)
* Atribute penghubung: NIP\_dosen(FK NIP\_dosen di ruang\_kelas)

1. Rancangan database mengenai Pembeli, Toko, Pemilik, dan Barang

Langkah-langkah perancangan database:

1. Menentukan entities:
2. Pembeli: menyimpan semua data pribadi pembeli.
3. Toko : menyimpan semua data tentang toko.
4. Pemilik : menyimpan semua data tentang pemilik.
5. Barang : menyimpan informasi semua tentang barang.
6. Menentukan atribute(sifat-sifat) masing-masing entiti sesuai kebutuhan database
7. Pembeli

* Id\_pembeli: id yang dimiliki pembeli (varchar(25)) PK
* Nama\_pembeli: nama lengkap pembeli (varchar(25))
* Alamat\_pembeli: alamat lengkap pembeli (varchar(255))

1. Toko

* Nama\_toko: nama toko (varchar(45))
* No\_toko: nomor toko (varchar(24)) PK
* Alamat: alamat lengkap toko (varchar(255))

1. Pemilik

* Nama\_pemilik: nama lengkap pemilik toko(varchar(25))
* Id\_pemilik: id yang dimiliki pemilik (varchar(25)) PK
* Alamat: alamat lengkap pemilik(varchar(225))

1. Barang

* Nama\_barang: nama barang(varchar(25))
* Kode\_barang: kode barang(varchar(25)) PK
* Jenis: jenis barang yang di jual(varchar(25))

1. Menentukan relationship (hubungan) antar entitas

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | pembeli | toko | pemilik | barang |
| pembeli | - | 1:1 | - | n:1 |
| toko |  | - | m:n | m:n |
| pemilik |  |  | - | - |
| barang |  |  |  | - |

Hubungan

* Pembeli mengunjungi toko:
* Tabel utama: pembeli, toko
* Tabel kedua: toko
* Relationship: one-to-one(1:1)
* Atribute pengubung:
* Pembeli membeli barang:
* Tabel utama: pembeli, barang
* Tabel kedua: pembeli\_has\_barang
* Relationship: one\_to\_many(1:n)
* Atribute penghubung: id\_pembeli(FK id\_pembeli di barang)
* Toko dimiliki pemilik:
* Tabel utama: toko, pemilik
* Tabel kedua: toko\_has\_pemilik
* Relationship: many\_to\_many(m:n)
* Atribute penghubung: id\_toko, id\_pemilik(FK id\_toko, id\_pemilik di toko\_has\_pemilik
* Toko penempatan barang
* Tabel utama: toko, barang
* Tabel kedua: toko\_has\_barang
* Relationship: many-to-many(m:n)
* Atribute penghubung: id\_toko, id\_barang (FK id\_toko, id\_barang di toko\_has\_barang