

**LAPORAN PRAKTIKUM  
BASIS DATA  
PRAKTIKUM III**



**Disusun oleh :**  
PUPUT SURYA NINGTYAS  
V3523015

**Dosen**  
Masbahah, S.pd., M.pd.

**PSDKU D-III TEKNIK INFORMATIKA  
SEKOLAH VOKASI  
UNIVERSITAS SEBELAS MARET  
2024**

# **BAB 1**

## **PENDAHULUAN**

### **1.1 Latar Belakang**

Entity Relationship Diagram (ERD) merupakan alat penting dalam pemodelan data untuk menggambarkan hubungan antar entitas dalam database. ERD bagaikan cetak biru yang menuntun pembangunan database, layaknya arsitek yang merancang denah sebelum membangun rumah. ERD terdiri dari berbagai elemen, seperti entitas, atribut, dan relasi, yang ibarat batu bata dan semen dalam konstruksi database. Memahami elemen-elemen ini secara mendalam merupakan kunci untuk membangun database yang terstruktur, efisien, dan mudah dipahami. Praktikum ERD dirancang untuk membekali mahasiswa dengan pemahaman komprehensif tentang elemen-elemen kunci ERD. Dalam praktikum ini, mahasiswa akan mempelajari berbagai varian entity, atribut, dan relasi, serta cara mengaplikasikannya dalam merancang database untuk kasus yang spesifik.

### **1.2 Tujuan**

- Memahami konsep dasar ERD dan elemen-elemennya.
- Memahami konsep dasar ERD dan elemen-elemennya.
- Mampu mengidentifikasi dan memodelkan entity, atribut, dan relasi dalam ERD untuk kasus yang spesifik.

### **1.3 Manfaat**

- Memahami konseptualisasi data dalam database.
- Mampu menghasilkan ERD yang akurat dan efisien untuk kebutuhan database.
- Meningkatkan kemampuan berpikir logis dan sistematis dalam merancang database.

### **1.4 Alat dan Bahan**

- Computer/laptop
- Os (windows/linux)
- Draw io

## **BAB 2**

### **TINJAUAN PUSTAKA**

#### **2.1 ERD**

Entity Relationship Diagram (ERD) merupakan alat penting dalam pemodelan data untuk menggambarkan hubungan antar entitas dalam database. ERD terdiri dari berbagai elemen, seperti entitas, atribut, dan relasi, yang memiliki peran penting dalam merancang database yang terstruktur dan efisien. Memahami varian-varian elemen ini secara mendalam merupakan kunci untuk membangun database yang akurat, efisien, dan mudah dipahami.

#### **2.2 Varian Entity**

- Strong Entity: Entitas yang memiliki identitas unik dan tidak bergantung pada entitas lain. (Elmasri & Navathe, 2016)
- Weak Entity: Entitas yang tidak memiliki identitas unik dan bergantung pada entitas lain. (Kroenke, 2019)
- Subtype Entity: Entitas yang merupakan spesialisasi dari entitas lain. (Rob & Coronel, 2019)

#### **2.3 Varian Atribut**

- Simple Attribute: Atribut yang tidak terbagi lagi. (Elmasri & Navathe, 2016)
- Composite Attribute: Atribut yang terdiri dari beberapa atribut lain. (Kroenke, 2019)

#### **2.4 Varian Relasi**

- Unary Relation: Relasi yang menghubungkan entitas dengan dirinya sendiri. (Rob & Coronel, 2019)
- N-ary Relation: Relasi yang menghubungkan lebih dari dua entitas. (Elmasri & Navathe, 2016)
- Redundant Relation: Relasi yang memiliki informasi yang sama dengan relasi lain. (Kroenke, 2019)

## BAB 3

### HASIL DAN PEMBAHASAN

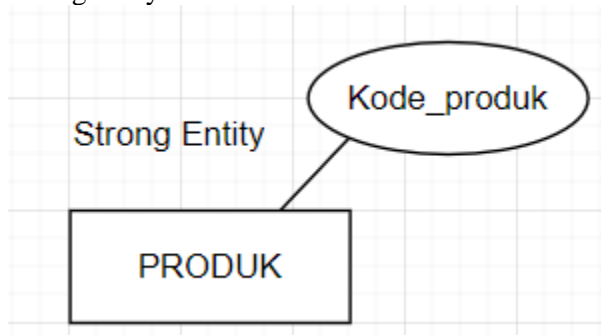
#### Latihan Soal

1. Sebutkan 2 contoh pada masing-masing varian entity. Jelaskan!
  - a. Strong entity
  - b. Weak entity
  - c. Subtype entity
2. Sebutkan 2 contoh pada masing-masing varian atribut. Jelaskan!
3. Sebutkan 2 contoh pada masing-masing varian relasi. Jelaskan!
  - a. unary relation
  - b. N-ary Relation
  - c. Redudant Relation
4. Silahkan analisis lebih lanjut ERD yang Anda kembangkan sesuai dengan final design

#### Jawaban

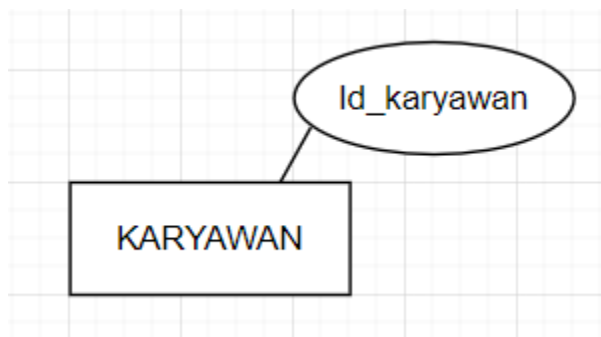
##### 1. Varian entity

- a. Strong entity



Penjelasan :

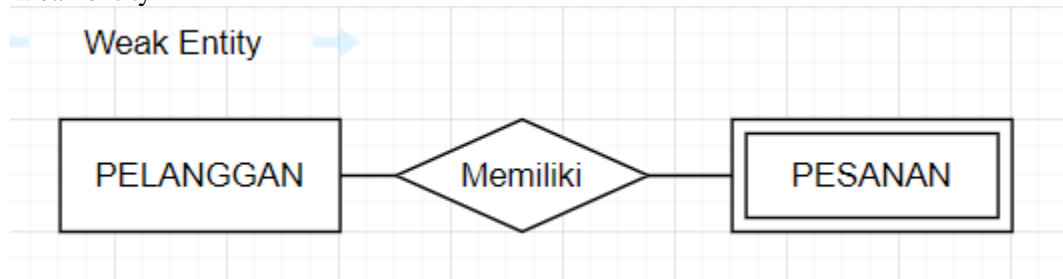
Keberadaan entitas Produk tidak bergantung pada entitas lain.



Penjelasan :

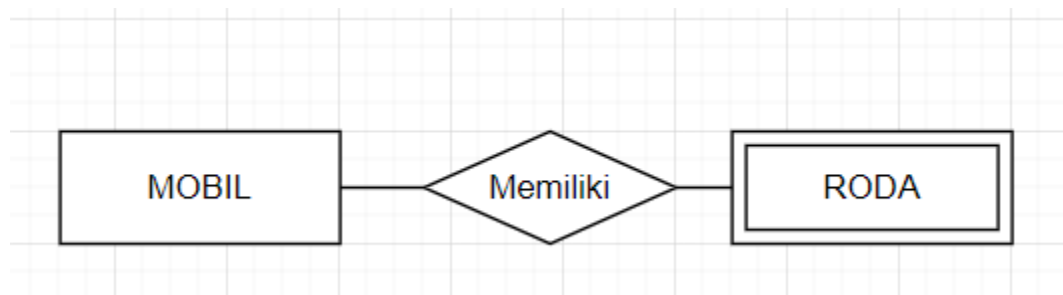
Keberadaan entitas Karyawan tidak bergantung pada entitas lain.

b. Weak entity



Penjelasan:

Keberadaan entitas Pesanan bergantung pada entitas Pelanggan. Sebuah Pesanan harus memiliki ID pelanggan yang valid.

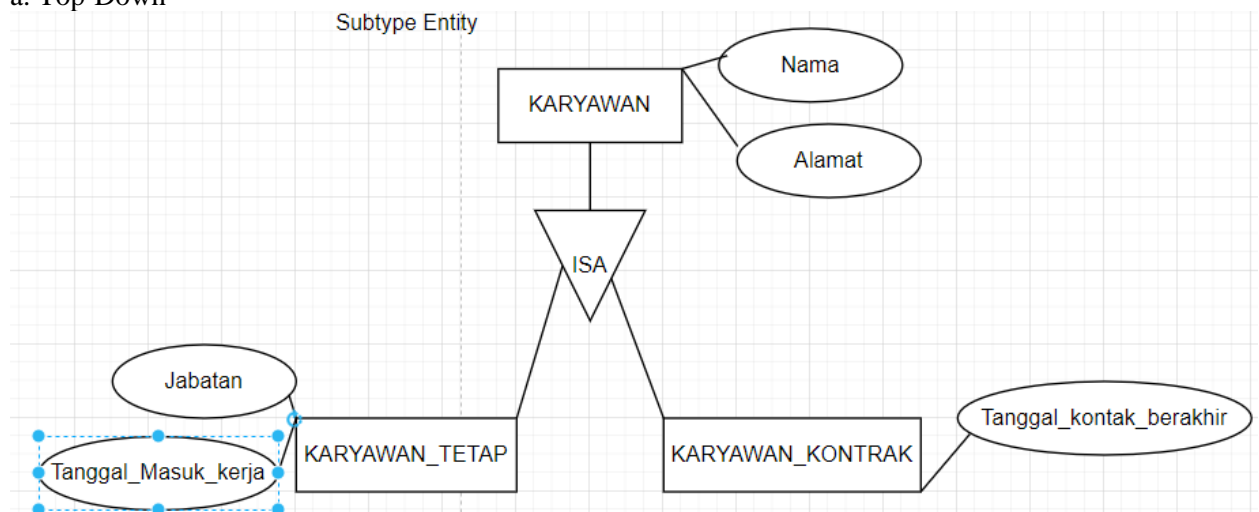


Penjelasan:

Bagian dari sebuah mobil adalah contoh dari weak entity karena bagian tersebut tidak dapat diidentifikasi secara unik dalam basis data tanpa mengacu pada mobil sebagai entitas pemilik.

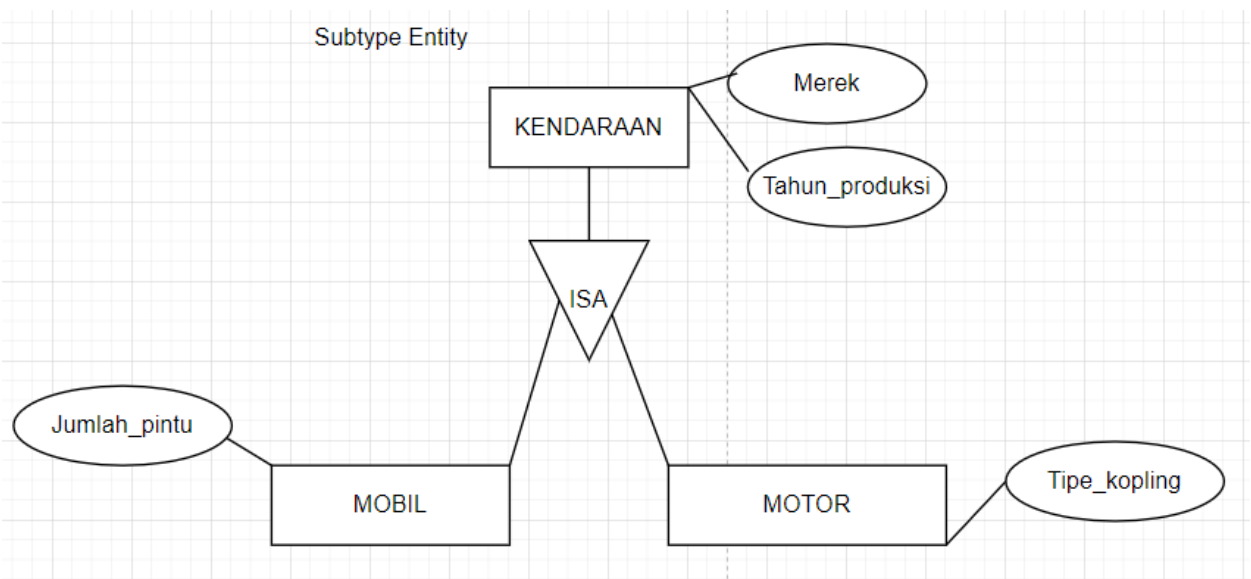
c. Subtype entity

a. Top-Down



Penjelasan:

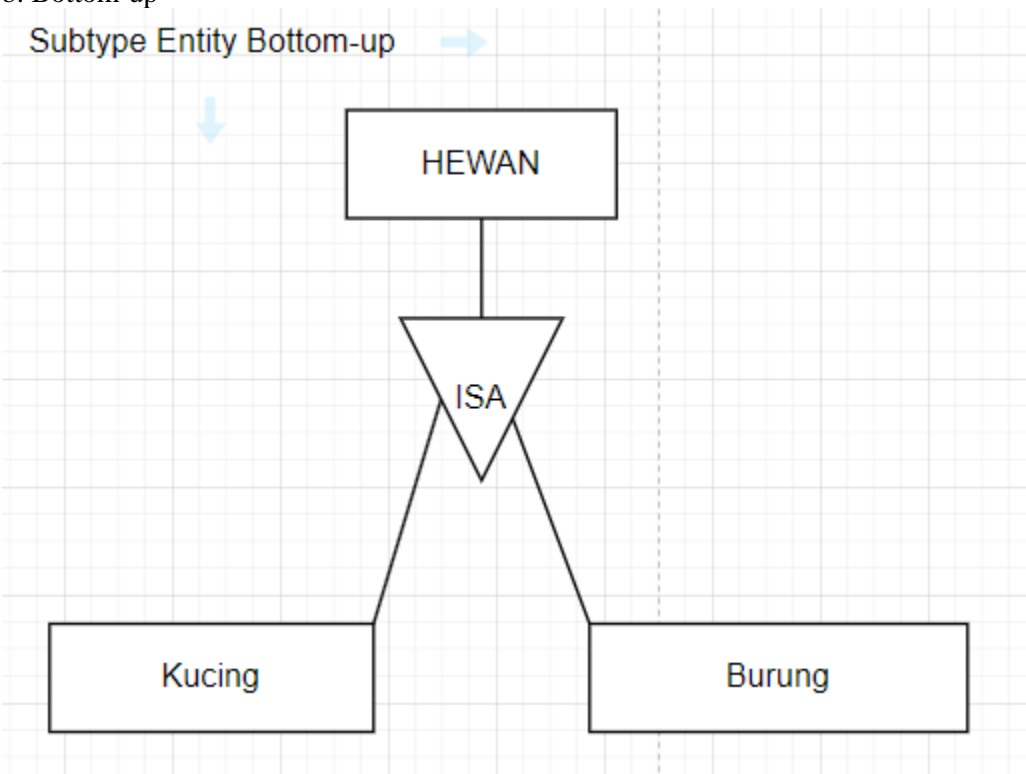
Karyawan Tetap dan Karyawan Kontrak adalah subtype dari entitas Karyawan. Mereka memiliki atribut yang sama (misalnya, nama, alamat) dan atribut tambahan yang spesifik untuk subtipenya tanggal kontrak berakhir untuk Karyawan Kontrak.



Penjelasan:

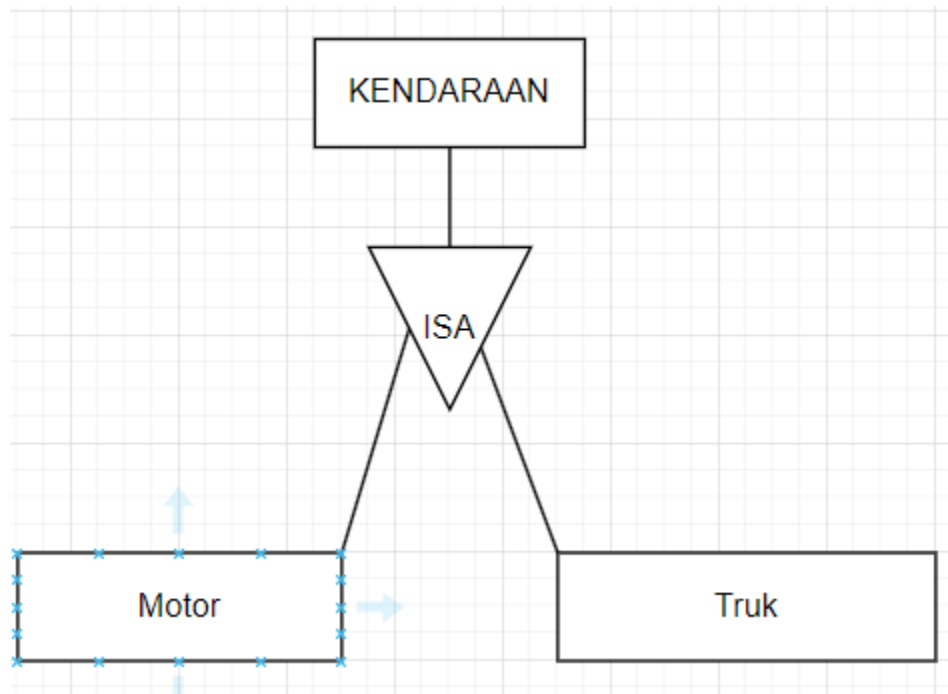
Mobil dan Motor adalah subtype dari entitas Kendaraan. Mereka memiliki atribut yang sama (misalnya, merek, tahun produksi) dan atribut tambahan yang spesifik untuk subtipenya (misalnya, jumlah pintu, tipe kopling).

b. Bottom-up



Penjelasan :

Hewan adalah entitas yang lebih umum, sedangkan Kucing dan Burung adalah jenis-jenis atau variasi yang lebih spesifik dari hewan.

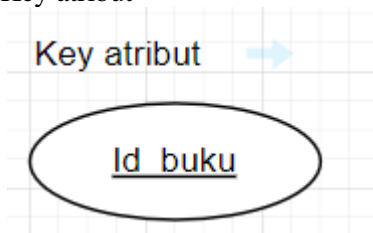


Penjelasan:

Kendaraan adalah entitas yang lebih umum, sedangkan Motor dan Truk adalah jenis-jenis atau variasi yang lebih spesifik dari kendaraan.

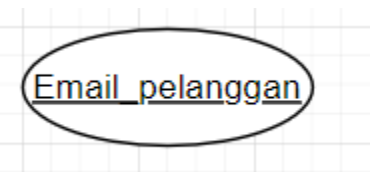
## 2. Varian atribut

### a. Key atribut



Penjelasan:

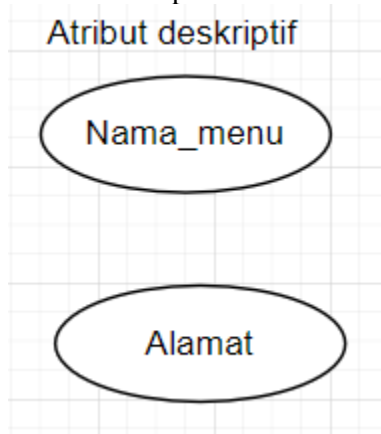
Dalam sebuah basis data untuk sistem manajemen perpustakaan, atribut "ID\_Buku" bisa menjadi kunci utama yang mengidentifikasi setiap buku secara unik.



Penjelasan:

Dalam sistem manajemen pelanggan untuk sebuah toko online, atribut "Email\_Pelanggan" bisa menjadi kunci utama karena setiap pelanggan memiliki alamat email yang unik.

b. Atribut deskriptif

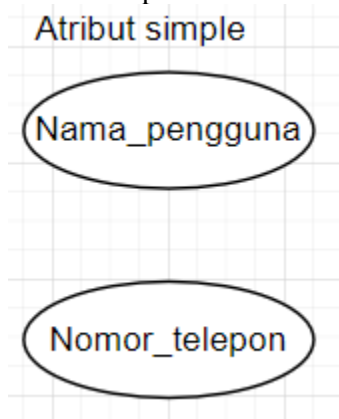


Penjelasan:

Contoh 1: Dalam sebuah sistem manajemen inventaris untuk sebuah restoran, atribut "Nama\_Menu" bisa digunakan untuk memberikan deskripsi tentang item menu.

Contoh 2: Dalam sistem manajemen karyawan sebuah perusahaan, atribut "Alamat" bisa menjadi atribut deskriptif untuk menggambarkan lokasi fisik karyawan tersebut.

c. Atribut simple



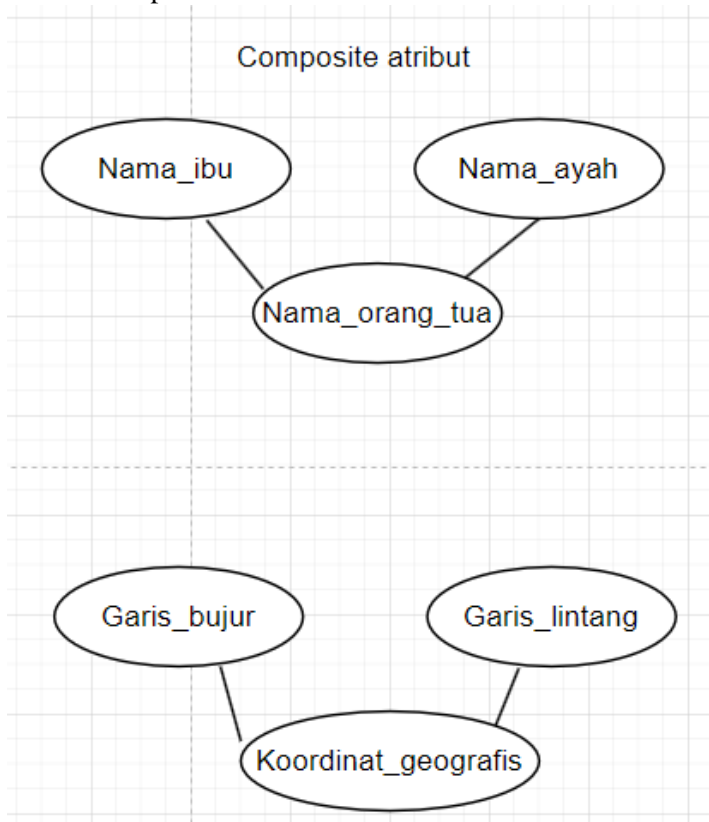
Penjelasan:

Contoh 1: Dalam sistem manajemen akun pengguna untuk sebuah platform media sosial, atribut "Nama\_Pengguna" hanya memuat satu nilai tunggal untuk setiap pengguna.

Contoh 2: Dalam basis data pelanggan sebuah toko, atribut "Nomor\_Telepon" hanya menyimpan satu nomor telepon untuk setiap pelanggan.



d. Atribut komposit



Penjelasan:

Contoh 1: Atribut "Nama Orang Tua" bisa terdiri dari "Nama Ayah" dan "Nama Ibu".

Contoh 2: Atribut "Koordinat Geografis" bisa terdiri dari dua bagian, yaitu "Garis Lintang" (latitude) dan "Garis Bujur" (longitude).

e. Atribut single valued

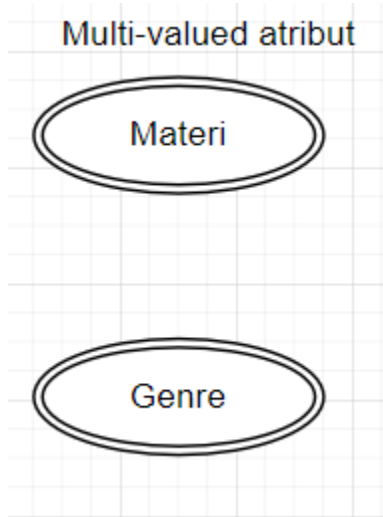


Penjelasan:

Contoh 1: Dalam sistem manajemen produk untuk sebuah toko online, atribut "Harga" hanya memiliki satu nilai untuk setiap produk.

Contoh 2: Dalam basis data mahasiswa sebuah perguruan tinggi, atribut "Tanggal\_Lahir" hanya memiliki satu nilai untuk setiap mahasiswa.

f. Atribut multi valued

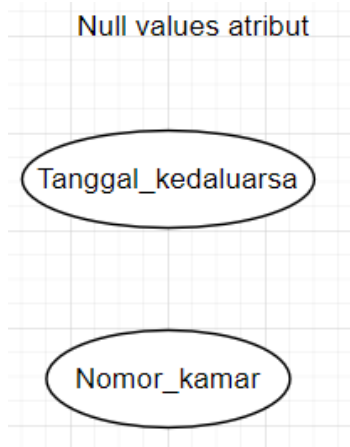


Penjelasan:

Contoh 1: Dalam sistem manajemen proyek untuk sebuah perusahaan konstruksi, atribut "Materi" dari sebuah proyek bisa memiliki beberapa nilai untuk mencakup berbagai jenis material yang digunakan.

Contoh 2: Dalam basis data film untuk sebuah platform streaming, atribut "Genre" memiliki beberapa nilai untuk menandai berbagai kategori genre film yang relevan.

g. Atribut null values

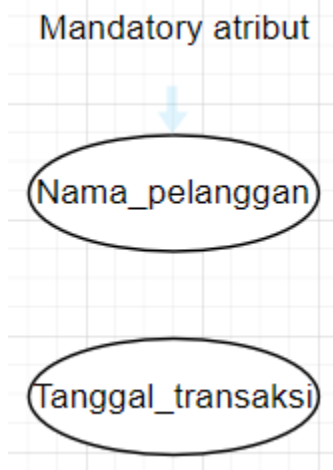


Penjelasan:

Contoh 1: Dalam sistem manajemen inventaris untuk sebuah gudang, atribut "Tanggal\_Kedaluarsa" memiliki nilai null untuk barang-barang yang tidak memiliki tanggal kedaluarsa.

Contoh 2: Dalam basis data karyawan sebuah perusahaan, atribut "Nomor\_Kamar" memiliki nilai null untuk karyawan yang belum ditempatkan di kamar tertentu.

h. Atribut mandatory

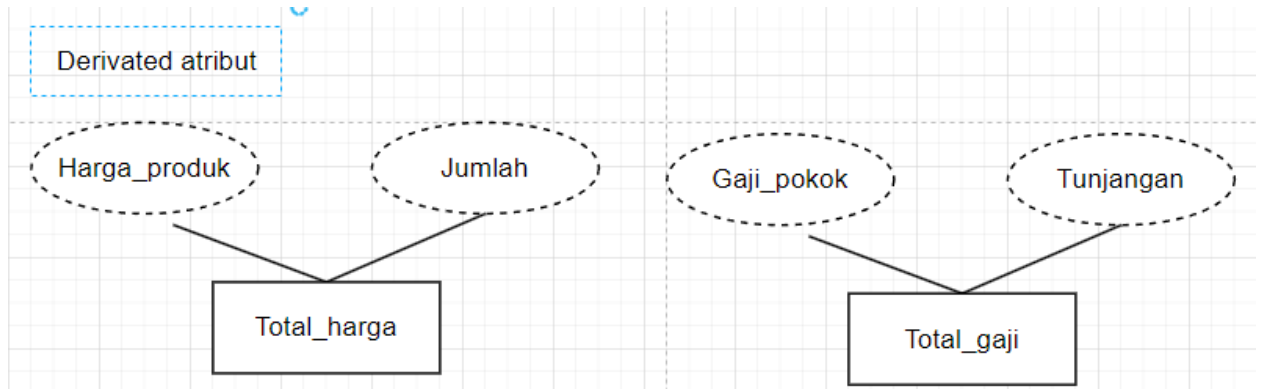


Penjelasan:

Contoh 1: Dalam sistem manajemen pesanan untuk sebuah restoran, atribut "Nama\_Pelanggan" menjadi atribut yang wajib diisi untuk setiap pesanan yang dibuat.

Contoh 2: Dalam basis data penjualan sebuah toko, atribut "Tanggal\_Transaksi" harus diisi untuk setiap catatan transaksi yang dimasukkan.

i. Atribut derived



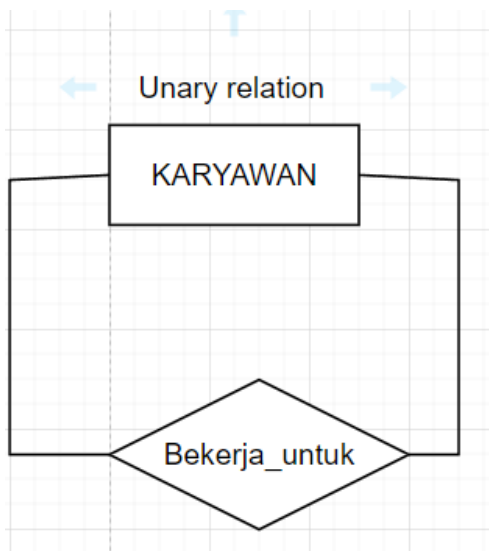
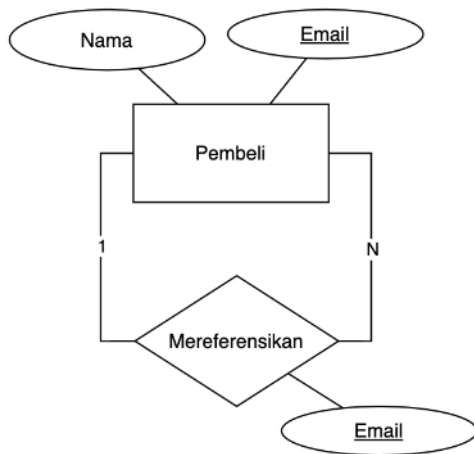
Penjelasan:

Contoh 1: Dalam sistem manajemen inventaris untuk sebuah toko pakaian, atribut "Total\_Harga" bisa dihitung secara otomatis dari perkalian antara atribut "Harga\_Produk" dan "Jumlah".

Contoh 2: Dalam sistem manajemen karyawan sebuah perusahaan, atribut "Total Gaji" bisa dihitung dari gaji pokok ditambah dengan tunjangan dan bonus yang diterima karyawan. Jadi, nilai "Total Gaji" adalah hasil penjumlahan dari beberapa atribut lain yang menyatakan komponen gaji.

### 3. Varian relasi

a. Unary relation

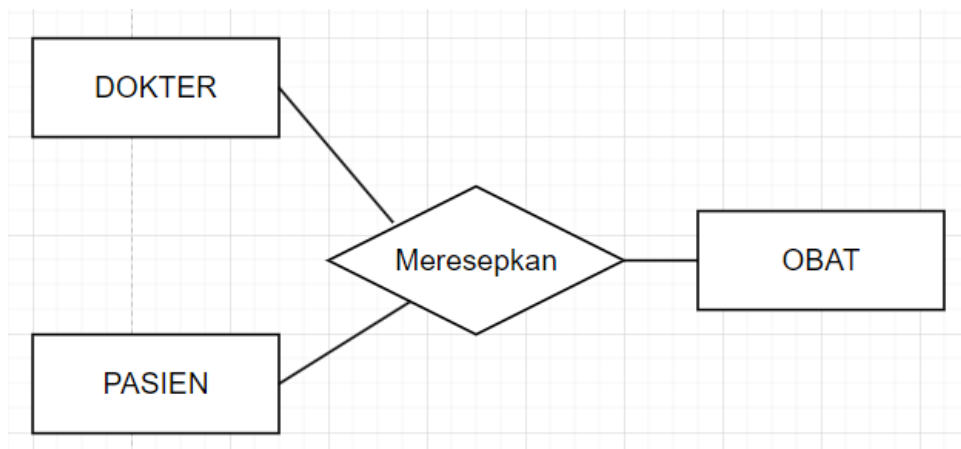
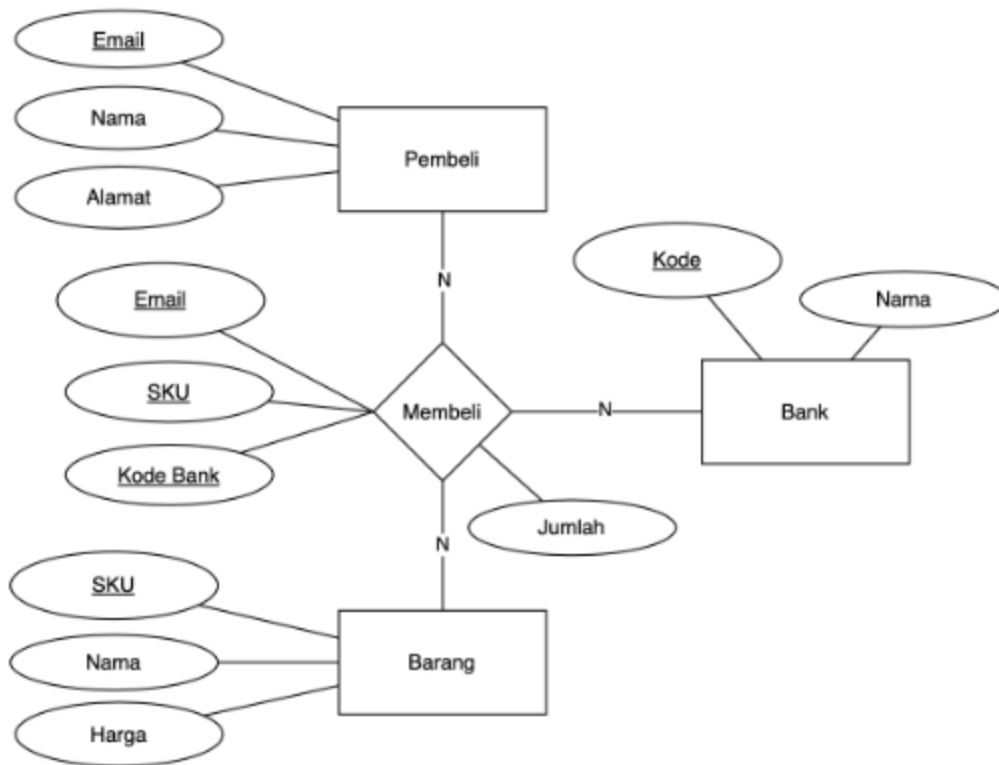


Penjelasan:

Contoh 1: Pada kasus Toko Online, misal kita akan menambah fitur Referral (Member get Member), dan kita akan membolehkan Pembeli merekomendasikan Pembeli lain untuk mendapat bonus berupa saldo tertentu

Contoh 2: Entitas Karyawan memiliki hubungan dengan dirinya sendiri. Hal ini dapat digunakan untuk memodelkan situasi di mana seorang karyawan dapat memiliki atasan (karyawan lain) di departemen yang sama.

b. N-ary relation

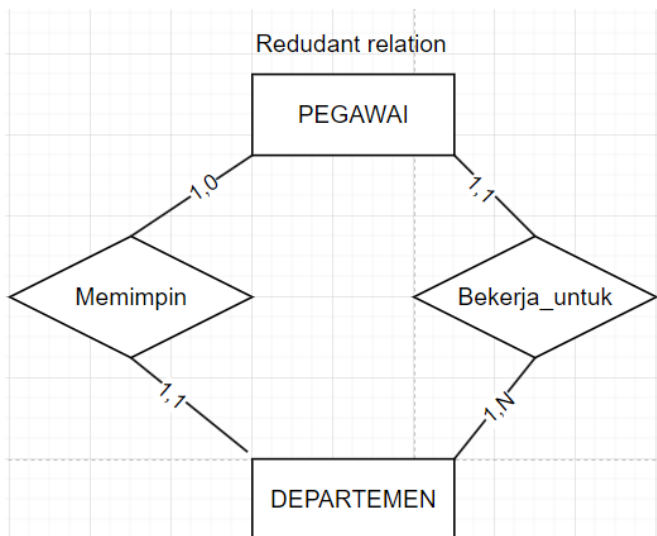
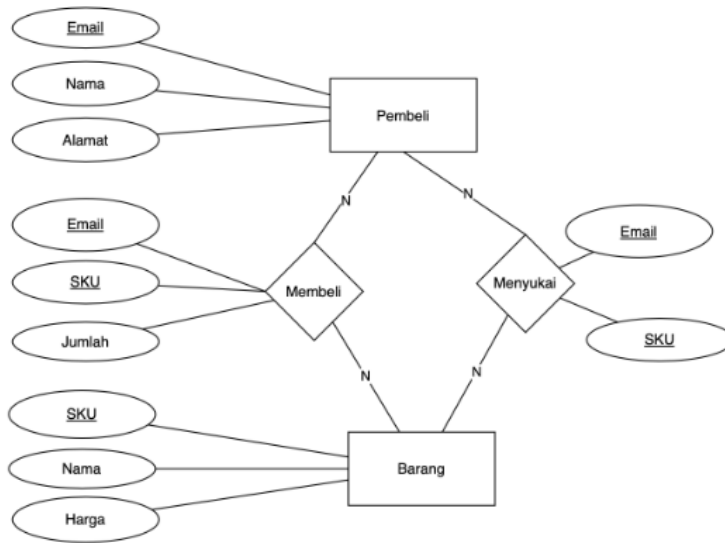


Penjelasan:

Contoh 1: Kita akan menambahkan Bank dengan relasi Pembeli membeli Barang, menggunakan Bank

Contoh 2: Dokter, pasien, dan obat. Seorang dokter dapat mereseapkan satu atau lebih obat kepada satu atau lebih pasien. Seorang pasien dapat memiliki satu atau lebih obat yang diresepkan oleh dokter.

c. Redundant relation

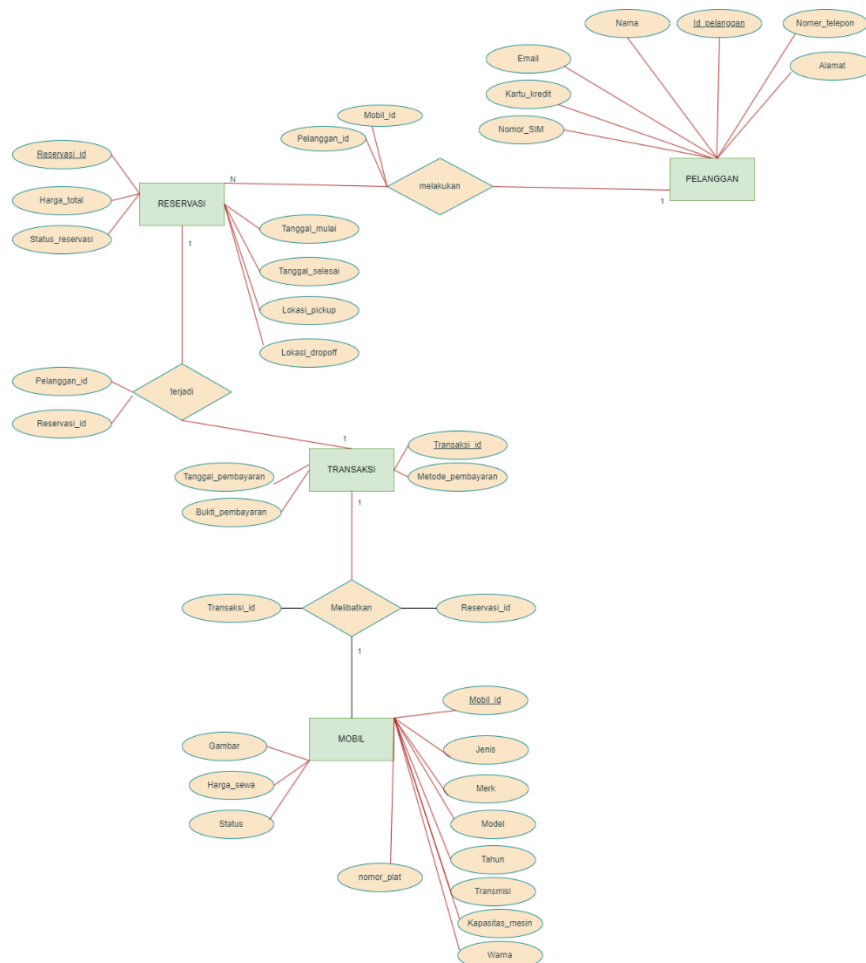


Penjelasan:

Contoh 1: Pada kasus relasi Pembeli dan Barang, misal kita tambahkan relasi Pembeli menyukai Barang, artinya tidak hanya Membeli, tapi juga Menyukai

Contoh 2: Kardinalitas 1,1 menunjukkan bahwa setiap Pegawai memimpin satu Departemen dan setiap Departemen dipimpin oleh satu Pegawai. kardinalitas 1,n menunjukkan bahwa setiap Pegawai bekerja untuk satu Departemen, tetapi satu Departemen dapat memiliki banyak Pegawai.

## 4. Analisis ERD



Mobil:

**Atribut:**

mobilID (primary key)

jenisMobil (atribut deskriptif)

merk (atribut deskriptif)

model (atribut deskriptif)

tahun (atribut deskriptif)

transmisi (atribut deskriptif)

kapasitasMesin (atribut deskriptif)

warna (atribut deskriptif)

nomorPlat (atribut deskriptif)

status (atribut deskriptif)

gambar (atribut deskriptif)

hargaSewa (atribut deskriptif)

**Penyewa:****Atribut:**

penyewaID (primary key)  
nama (atribut deskriptif)  
nomorTelepon (atribut multi value)  
email (atribut deskriptif)  
alamat (atribut deskriptif)  
nomorSIM (atribut deskriptif)  
kartuKredit (atribut deskriptif)

**Reservasi:****Atribut:**

reservasiID (primary key)  
penyewaID (foreign key)  
mobilID (foreign key)  
tanggalMulai (atribut deskriptif)  
tanggalSelesai (atribut deskriptif)  
lokasiPickup (atribut deskriptif)  
lokasiDropoff (atribut deskriptif)  
hargaTotal (derived atribut-> dihitung berdasarkan harga sewa, tanggal mulai, tanggal selesai)  
statusReservasi (atribut deskriptif)

**Transaksi:****Atribut:**

transaksiID (primary key)  
reservasiID (foreign key)  
tanggalPembayaran (atribut deskriptif)  
metodePembayaran (atribut deskriptif)  
buktiPembayaran (atribut deskriptif)



## **BAB 4**

### **KESIMPULAN**

#### **4.1 Kesimpulan**

Praktikum ERD varian atribut, relasi, entitas bermanfaat untuk meningkatkan pemahaman dan kemampuan dalam merancang database yang optimal. Dengan memahami dan mengimplementasikan berbagai jenis atribut, relasi, dan entitas, dapat dibuat diagram ERD yang lengkap dan akurat untuk berbagai studi kasus.

#### **4.2 Saran**

Untuk memaksimalkan pembelajaran dari praktikum ERD varian atribut, relasi, entitas, sebaiknya kita tingkatkan kompleksitas studi kasus. Ini akan mengasah kemampuan mengidentifikasi dan memodelkan elemen ERD yang lebih rumit. Selain itu, penggunaan alat bantu pemodelan ERD dapat mempercepat dan mempermudah proses pembuatan diagram. Setelah diagram selesai, lakukan simulasi dan pengujian untuk memastikan kebenaran dan efektivitasnya dalam mendukung kebutuhan database. Terakhir, pelajari lebih dalam tentang normalisasi database. Normalisasi membantu mengurangi redundansi dan meningkatkan efektivitas database yang dirancang. Dengan menerapkan semua saran ini, pemahaman kita tentang ERD akan semakin mendalam dan kemampuan merancang database pun akan semakin optimal.

## **DAFTAR PUSTAKA**

Elmasri, R., & Navathe, S. B. (2016). Fundamentals of database systems (7th ed.). Boston, MA: Pearson.

Kroenke, D. (2019). Database processing: Fundamentals, design, and implementation (14th ed.). Boston, MA: Pearson.

Rob, P., & Coronel, C. (2019). Database systems: Design, implementation, and management (12th ed.). Boston, MA: Cengage Learning.