

PRAKTIKUM 4

Materi:

Transformasi ERD ke Tabel Relasional.

Tujuan Praktikum:

Mahasiswa mampu memetakan diagram ER dan EER ke skema relasional.

Referensi:

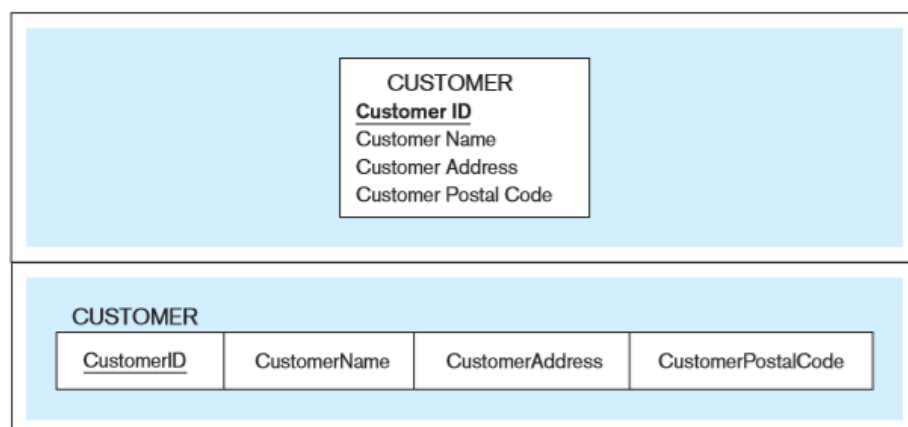
Hoffer, Jeffrey A., Mary B. Prescott and Fred R. McFadden. Ed-11. 2013. “Modern Database Management”. Prentice Hall.

PENYAJIAN:

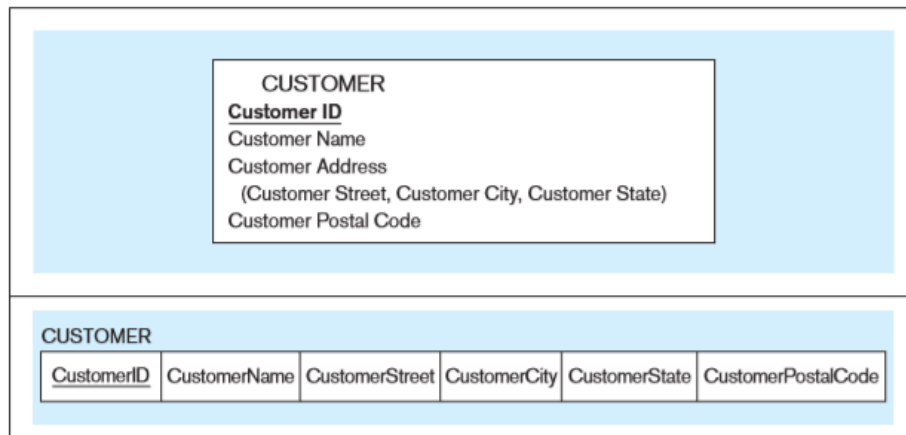
Relational data model merepresentasikan data dalam bentuk tabel. Dalam tahap *logical design*, kita mengubah ER (dan EER) diagram yang telah dikembangkan selama tahap *conceptual design* ke skema relasional.

Step 1: Map Regular Entities

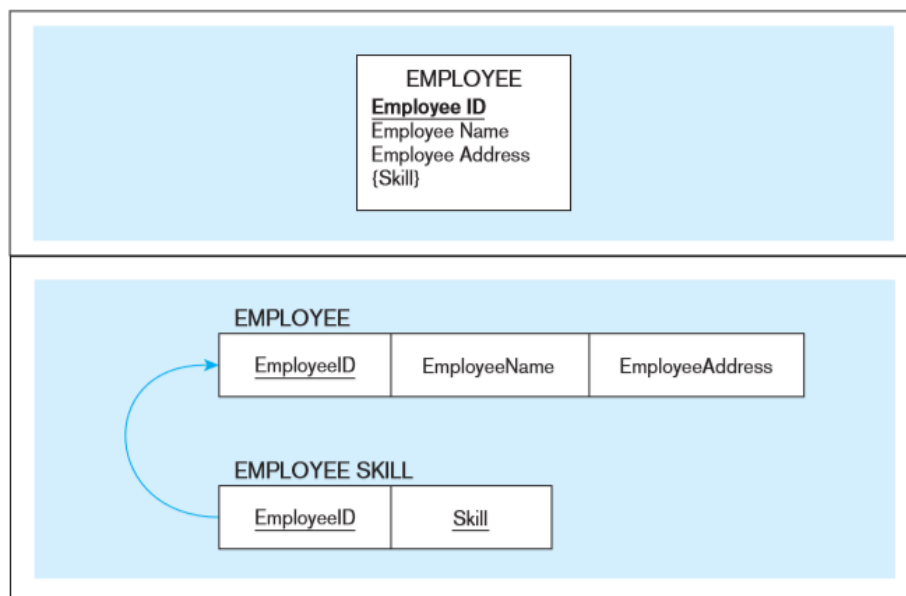
Setiap entitas dalam ER diagram diubah menjadi sebuah relasi. Nama relasi yang diberikan, sama dengan nama entitas. Setiap atribut pada entitas juga menjadi atribut pada relasi. Identifier pada entitas menjadi *primary key* pada relasi yang sesuai.



COMPOSITE ATTRIBUTES Jika sebuah entitas memiliki *composite attribute*, maka setiap komponen dari *composite attribute* akan menjadi atribut pada relasi yang baru.

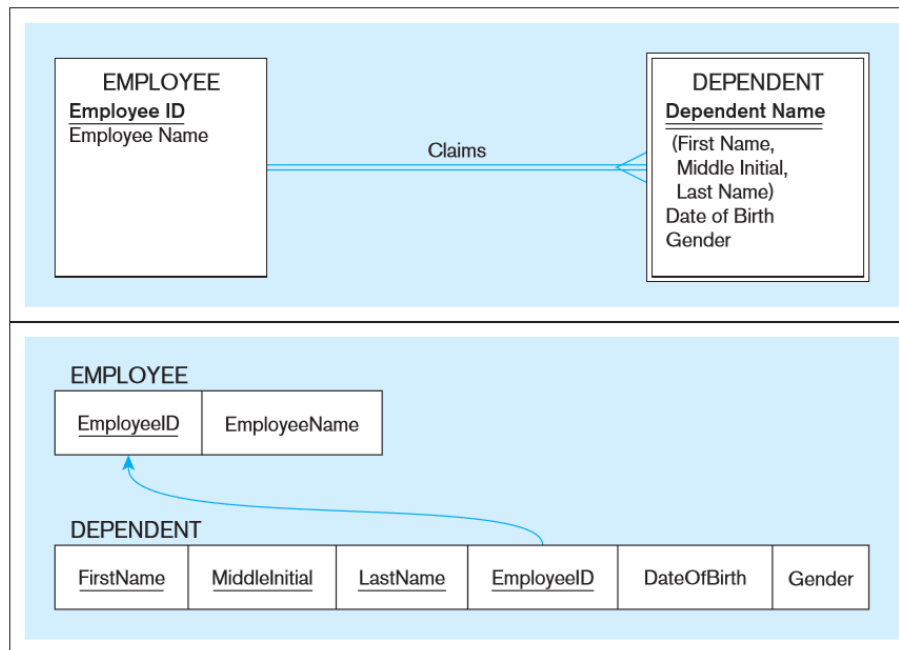


MULTIVALUED ATTRIBUTES Apabila entitas memiliki *multivalued attribute*, kita membuat dua relasi yang baru. Relasi pertama berisi semua atribut dari entitas tanpa *multivalued attribute*. Relasi kedua berisi dua atribut. Atribut pertama merupakan *primary key* dari relasi pertama, yang menjadi *foreign key* pada relasi kedua dan atribut yang kedua adalah *multivalued attribute*. Nama dari relasi kedua harus dapat menunjukan makna dari *multivalued attribute*.



Step 2: Map Weak Entities

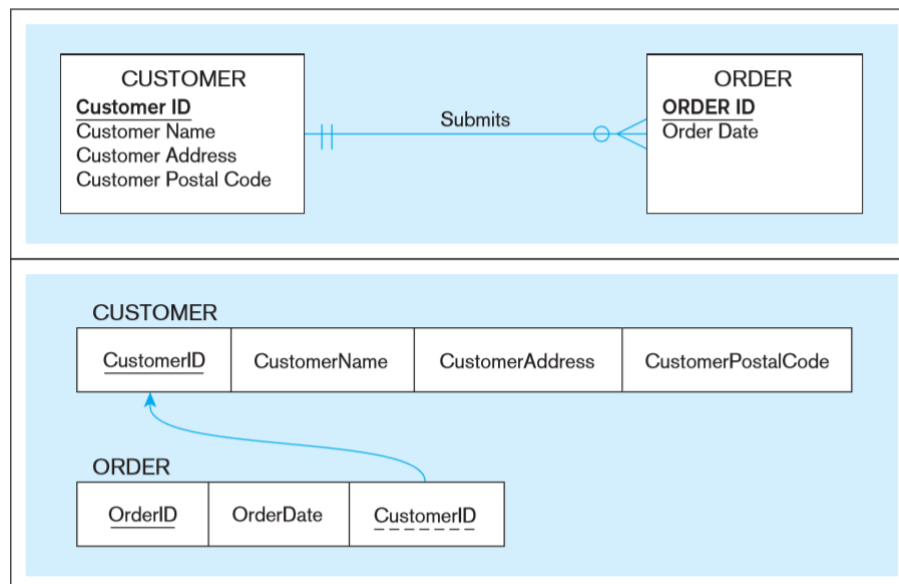
Untuk setiap entitas lemah, buat sebuah relasi baru dan masukan semua atribut entitas. Tambahkan *primary key* dari entitas kuat sebagai atribut *foreign key* pada relasi yang baru.



Step 3: MAP BINARY RELATIONSHIPS

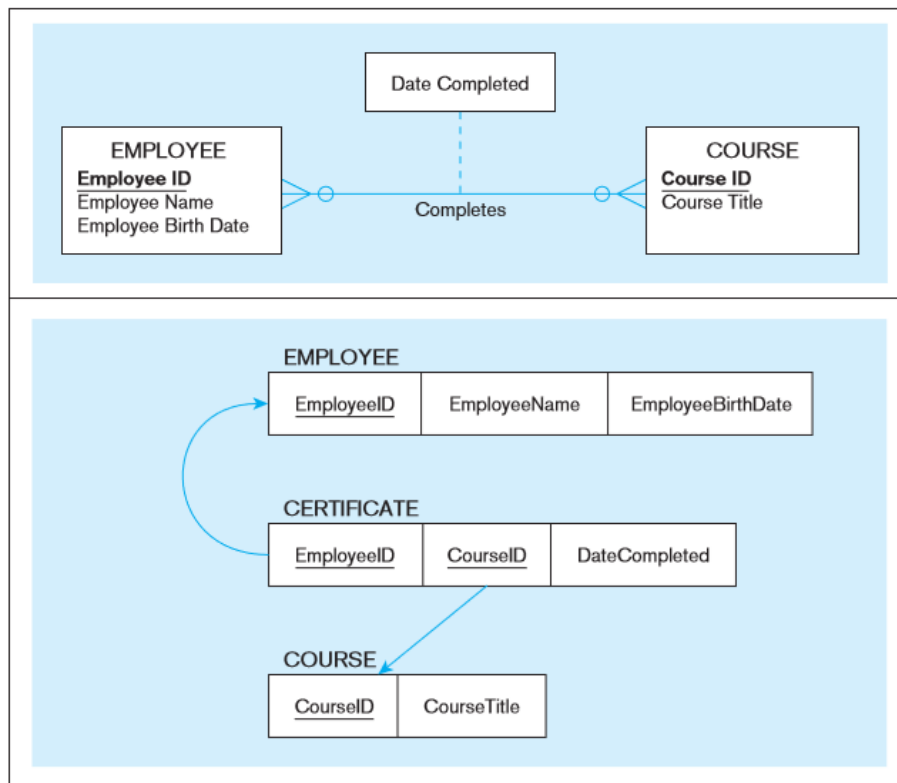
ONE-TO-MANY RELATIONSHIPS

Untuk 1:M relationship, buat relasi untuk setiap entitas dalam *relationship*. Tambahkan *primary key* dari *one-side* sebagai atribut *foreign key* pada relasi *many-side*.



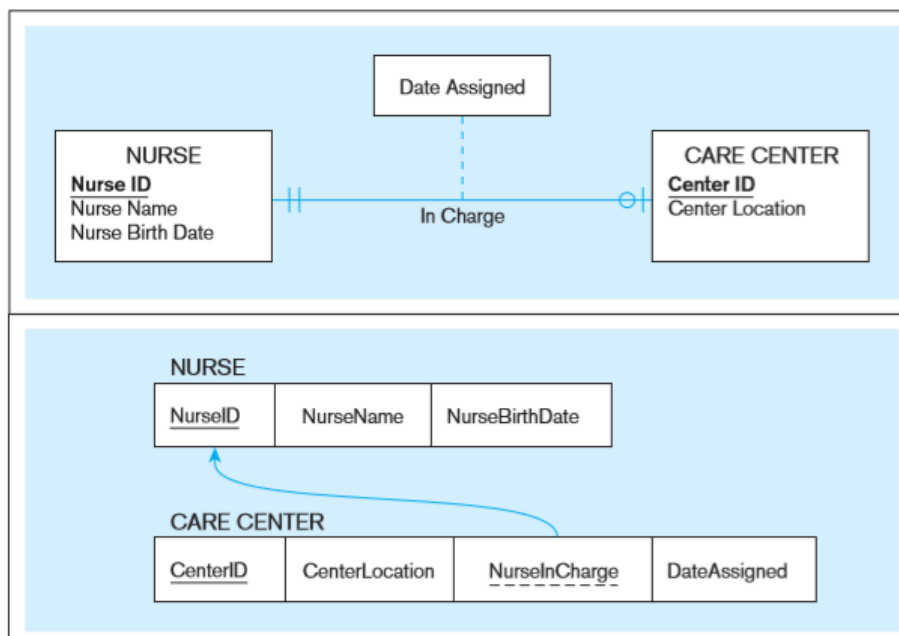
MANY-TO-MANY RELATIONSHIPS

Untuk M:N *relationship*, buat relasi untuk setiap entitas (A dan B) dalam *relationship*. Lalu buat relasi baru lagi (C) yang berisi atribut *primary key* dari A dan B. Kedua atribut ini menjadi *primary key* dari relasi C.



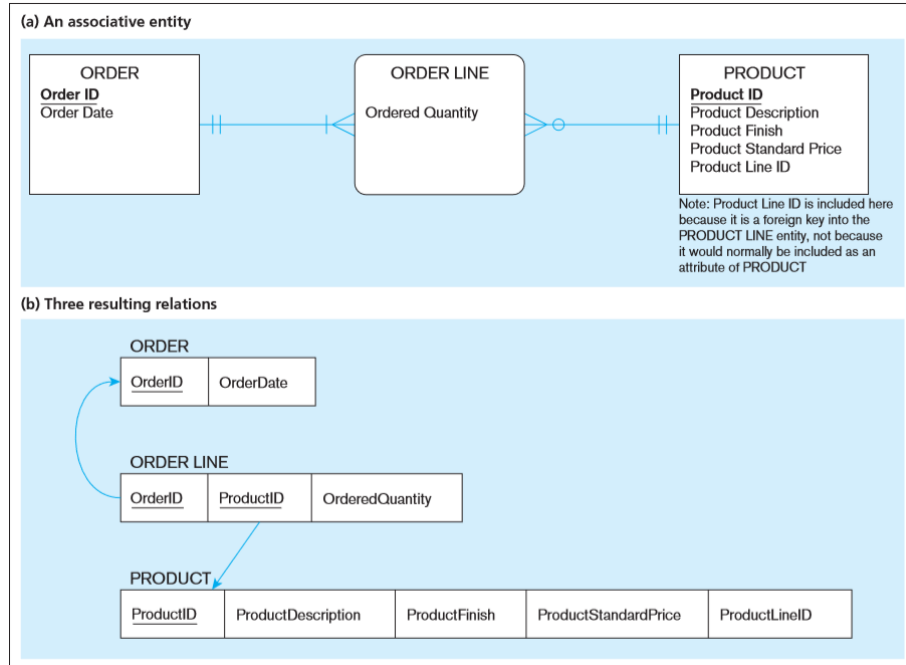
ONE-TO-ONE RELATIONSHIPS

Untuk setiap relasi biner 1:1, tambahkan *primary key* dari entitas yang lebih “berat” ke entitas yang lebih “ringan” sebagai *foreign key*. Entitas dianggap lebih berat apabila memiliki partisipasi total. Apabila ada atribut relasi maka tambahkan atribut tersebut pada entitas yang lebih “ringan”.

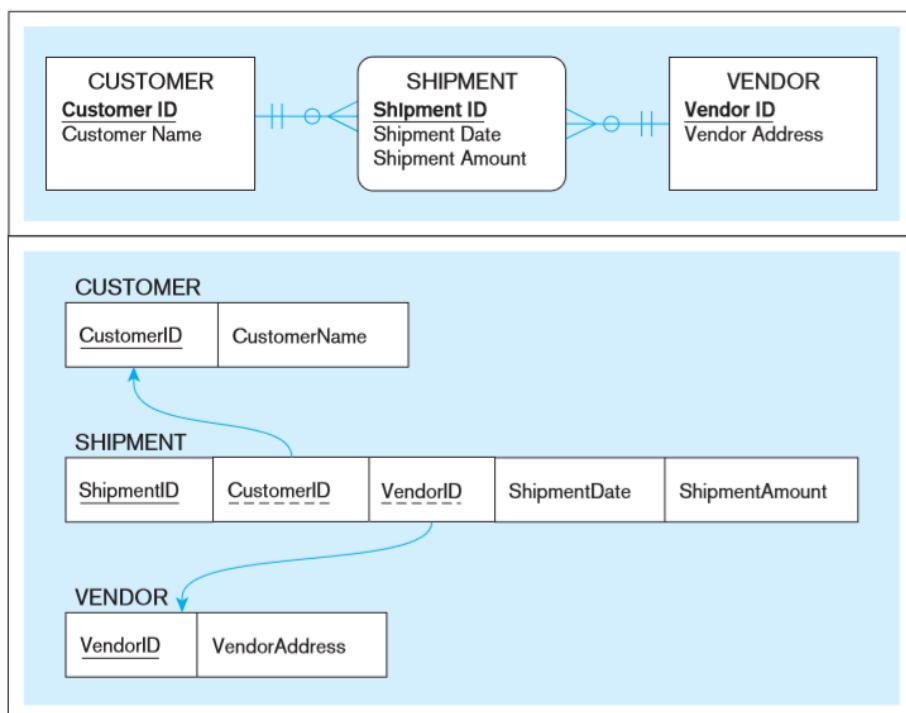


Step 4: MAP ASSOCIATIVE ENTITIES

IDENTIFIER NOT ASSIGNED Default primary key dari relasi associative terdiri dari dua atribut primary key milik dua relasi yang lain.

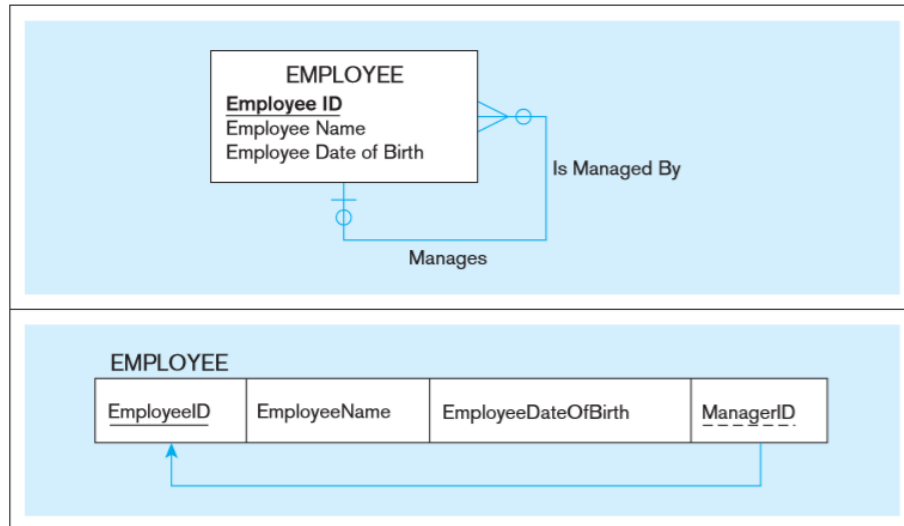


IDENTIFIER ASSIGNED Primary key dari relasi associative adalah identifier yang ada pada ER diagram (bukan default key). Primary key dari dua relasi yang lain, ditambahkan sebagai foreign key pada relasi associative.

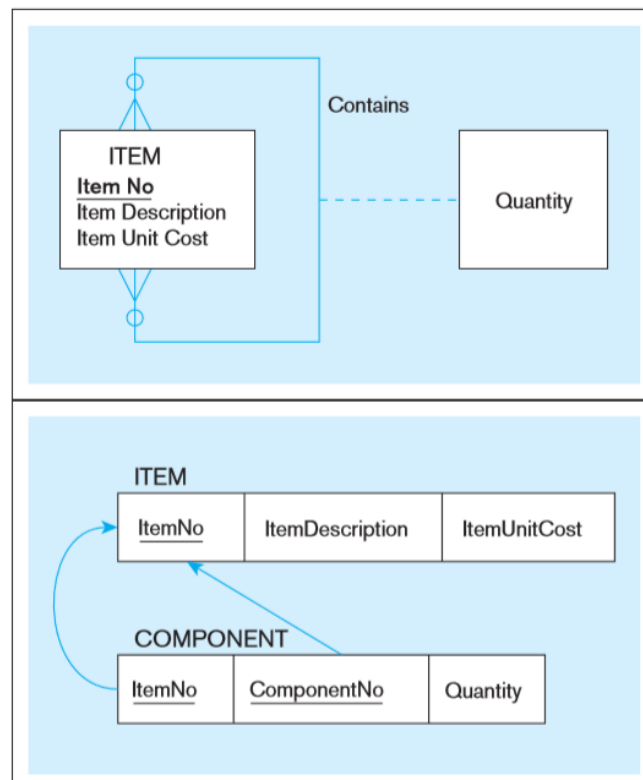


Step 5: MAP UNARY RELATIONSHIPS**UNARY ONE-TO-MANY RELATIONSHIPS**

Atribut *foreign key* ditambahkan ke relasi yang sama (atribut ini mereferensikan *primary key* pada relasi yang sama). Tipe *foreign key* ini disebut *recursive foreign key*.

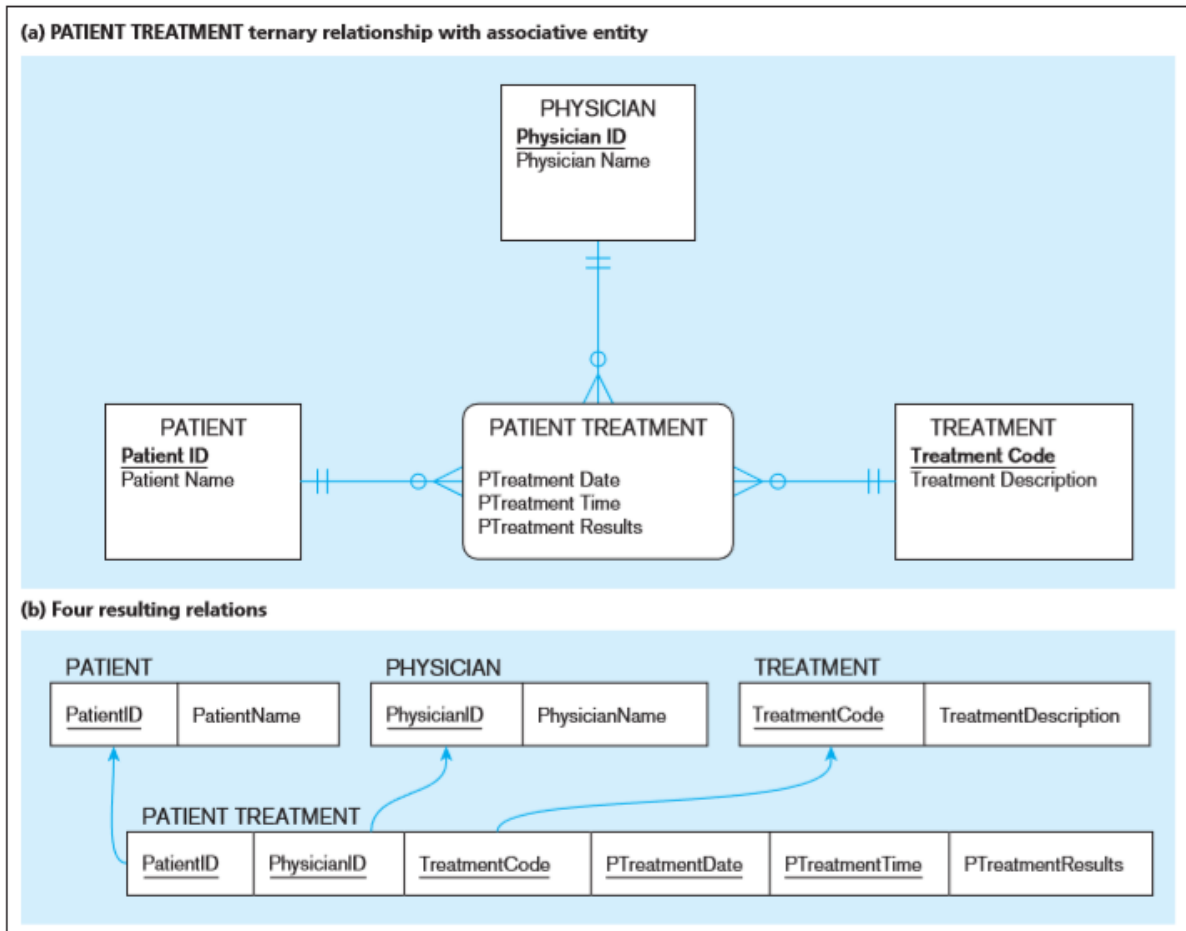
**UNARY MANY-TO-MANY RELATIONSHIPS**

Buat dua relasi baru: satu untuk merepresentasikan entitas dan satu lagi untuk relasi *associative*. *Primary key* dari relasi *associative* terdiri dari dua atribut. Kedua atribut ini diambil dari *primary key* relasi entitas (tidak boleh menggunakan nama yang sama).



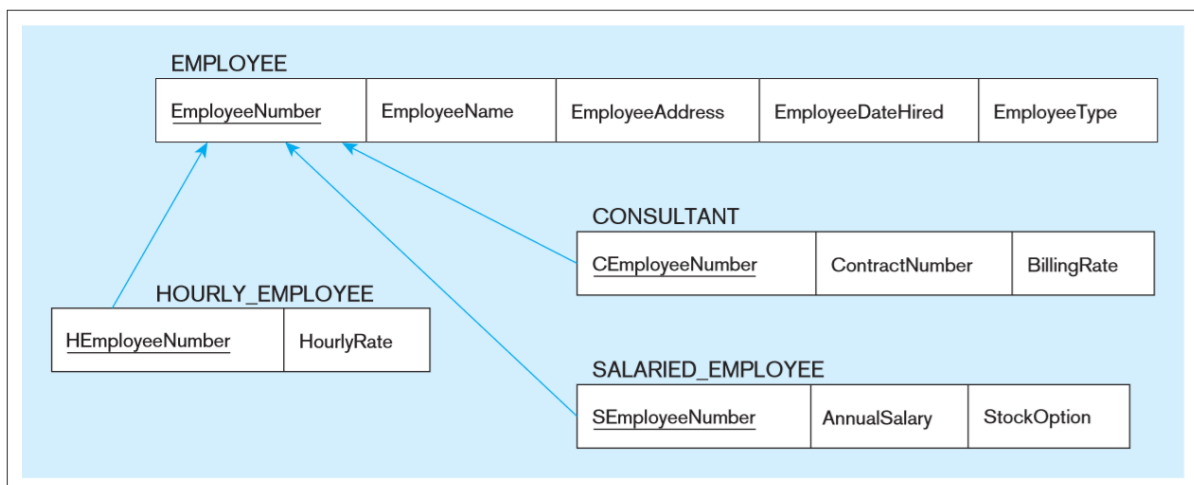
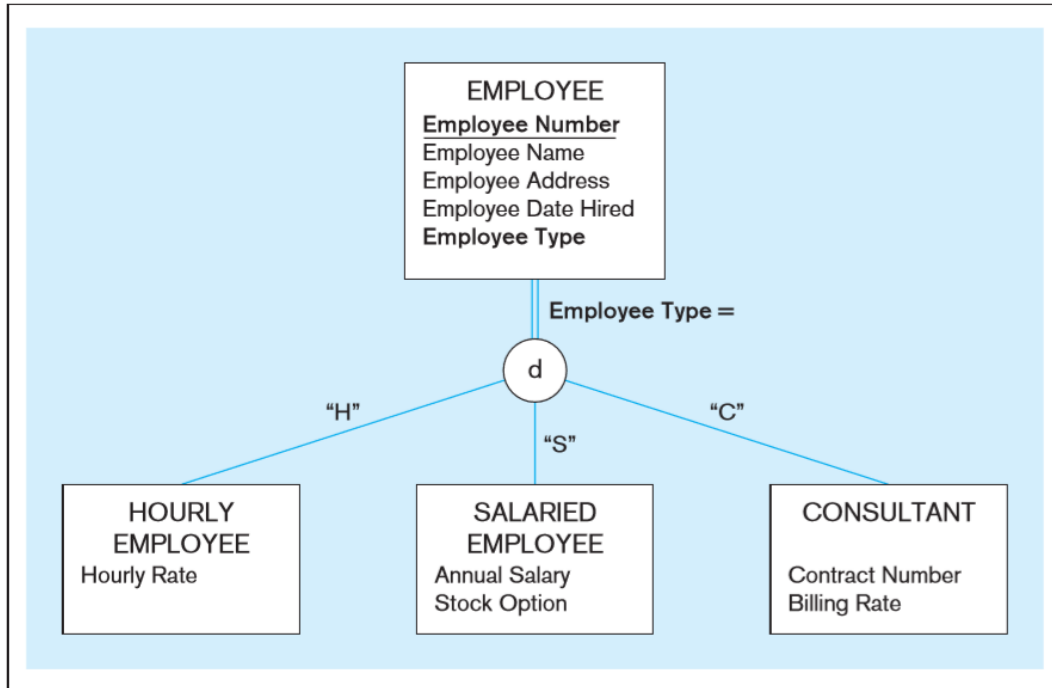
Step 6: MAP TERNARY (AND N-ARY) RELATIONSHIPS

Untuk setiap relasi n-ary dengan $n > 2$, buatlah tabel yang menyertakan seluruh *primary key* dari setiap entitas yang berelasi serta tambahkan atribut relasinya (jika ada).



Step 7: MAP SUPERTYPE/SUBTYPE RELATIONSHIPS

Buat relasi untuk superclass yang memiliki *primary key* dan *common atribut* yang dimiliki oleh setiap *subclass*. Lalu buat relasi terpisah untuk setiap *subclass* dengan *primary key* dan atribut spesifik yang berhubungan dengan *subclass* tersebut.



LEMBAR KERJA PRAKTIKUM

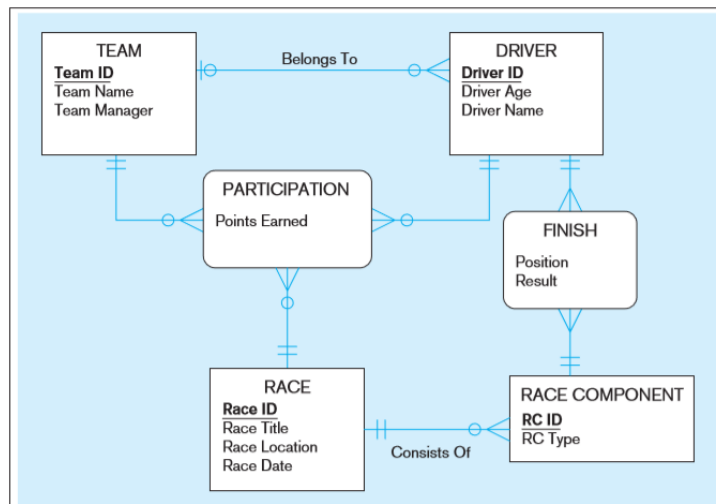
Nama:	Tanggal Praktikum:
NIM:	Waktu Praktikum:
Nilai:	Nama Asisten:

Soal:

1. Buatlah ER (dan EER) diagram pada contoh kasus di bawah, lalu lakukan transformasi menjadi skema relasional!

TomKat Entertainment adalah bisnis di bidang bioskop. Pemiliknya ingin membuat database untuk mengetahui film yang sedang atau sudah diputar di setiap studio bioskop. Setiap bioskop (diidentifikasi oleh Theater ID dan dideskripsikan dengan nama bioskop dan lokasi) memiliki satu atau banyak studio untuk memutar berbagai film. Setiap studio diidentifikasi oleh nomer dan dideskripsikan dengan kapasitas tempat duduk untuk menonton. Film dijadwalkan untuk diputar pada time slot tertentu. Setiap studio bisa memiliki time slot yang berbeda setiap harinya (tidak semua studio di satu bioskop memiliki waktu mulai yang sama dan pada hari yang berbeda satu film bisa diputar pada waktu yang berbeda di studio yang sama). Untuk setiap time slot, pemilik bioskop juga ingin mengetahui waktu selesai, jumlah pengunjung yang datang, dan harga tiket yang dijual. Setiap film diidentifikasi oleh Movie ID dan dideskripsikan dengan judul, durasi, dan tipe (bisa berupa trailer, feature, atau commercial). Satu atau lebih film diputar pada setiap time slot. Pemiliknya juga ingin mengetahui urutan film yang diputar (misalnya dalam satu time slot akan ada dua trailer, diikuti oleh dua commercial, diikuti oleh satu feature film, dan ditutup dengan satu commercial lain).

2. Untuk Diagram ER berikut, lakukan pemetaan ke dalam bentuk skema relasional:
 - a.



b.

