

## Pertemuan 2

### Relasi ERD dalam SQL Data Modeller

<b>Tujuan Intruksional</b>	:
Pokok Bahasan ini mempelajari tentang penggunaan Relasi di dalam SQL Data Modeler	
<b>Kompetensi Yang Diharapkan</b>	:
Mahasiswa diharapkan memahami tentang penggunaan Relasi dalam SQL Data Modeler termasuk jenis-jenis relasi, dan pembagian relasi berdasarkan kardinalitas dan opsionalitas.	
<b>Waktu Pertemuan</b>	: 100 Menit

#### I. Pengertian ERD

ERD yang memiliki kepanjangan yaitu *Entitas Relationship Diagram* merupakan salah satu dari data model. Dengan di gunakanya notasi-notasi grafik menggambarkan model ERD jadi lebih mudah di buat dan mudah juga untuk di baca. Model ini juga dapat langsung di transformasikan kedalam bentuk tabelnya.

#### II. Relasi dalam SQL Data Modeler

##### A. Pengertian Relasi

Relasi merupakan hubungan yang terjadi pada suatu tabel dengan tabel lainnya yang mempresentasikan hubungan antar objek di dunia nyata dan berfungsi untuk mengatur operasi suatu database.

##### B. Jenis-jenis Relasi

Dalam Oracle SQL Data Modeler, relasi antar tabel dibagi berdasarkan dua hal, yaitu kardinalitas dan opsionalitas.

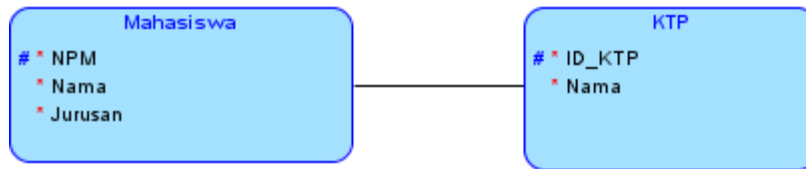
###### a. Kardinalitas (Cardinality)

Kardinalitas Relasi menunjukkan jumlah maksimum entitas yang dapat berelasi dengan entitas pada himpunan entitas yang lain. Kardinalitas relasi merujuk kepada hubungan maksimum yang terjadi dari himpunan entitas yang satu ke himpunan entitas yang lain dan begitu juga sebaliknya.

### 1) Satu ke satu (One to One)

Mempunyai pengertian "Setiap baris data pada tabel pertama dihubungkan hanya ke satu baris data pada tabel ke dua".

Contohnya antara mahasiswa dengan ktp.



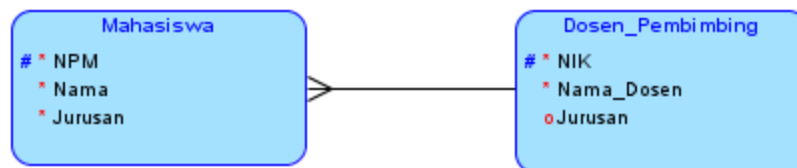
Cara membaca: satu mahasiswa harus memiliki satu ktp sedangkan ktp harus di miliki satu mahasiswa.

### 2) Satu ke Banyak / Banyak ke Satu (one to many / many to one)

Mempunyai pengertian "Setiap baris data dari tabel pertama dapat dihubungkan ke satu baris atau lebih data pada tabel ke dua".

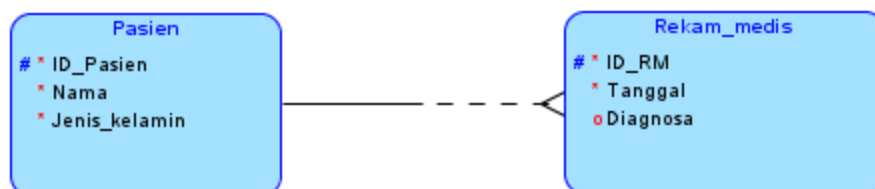
Contohnya :

#### 1. Relasi mahasiswa dibimbing oleh Dosen Pembimbing



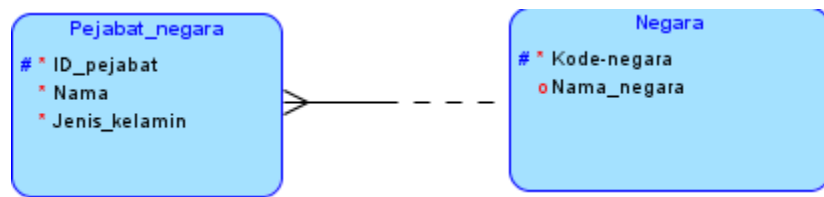
Cara membaca: Satu Dosen Pembimbing harus membimbing minimal satu mahasiswa atau lebih, dan setiap Mahasiswa harus memiliki satu dosen pembimbing.

#### 2. Relasi pasien memiliki Rekam medis



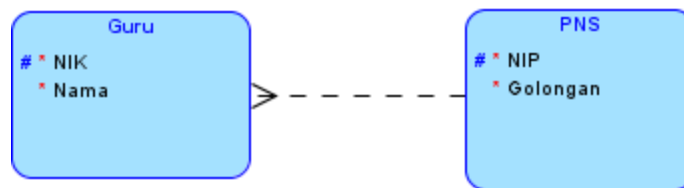
Cara membaca : Satu pasien dapat memiliki banyak rekam medis atau tidak sama sekali, tetapi satu rekam medis harus dimiliki oleh satu orang pasien.

### 3. Relasi Pejabat Negara memimpin Negara



Cara membaca : Satu negara dapat dipimpin oleh banyak pejabat negara, tetapi seorang pejabat negara tidak harus menjadi pemimpin negara.

### 4. Relasi Guru berstatus PNS



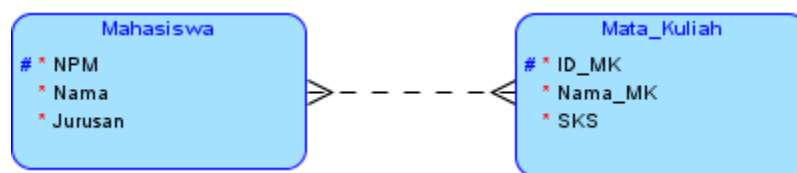
Cara membaca: Suatu status PNS dapat dimiliki oleh banyak guru, tetapi tidak semua guru memiliki status PNS.

## 3) Banyak ke Banyak (Many to Many).

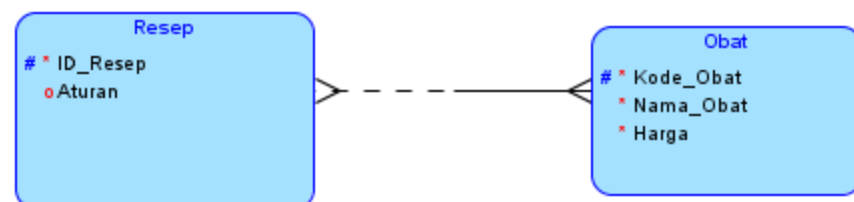
Mempunyai pengertian "Satu baris atau lebih data pada tabel pertama bisa dihubungkan ke satu atau lebih baris data pada tabel ke dua". Artinya ada banyak baris di tabel satu dan tabel dua yang saling berhubungan satu sama lain.

Contohnya :

1. Relasi antar tabel mahasiswa dan tabel mata kuliah. Satu baris mahasiswa bisa berhubungan dengan banyak baris mata pelajaran begitu juga sebaliknya.



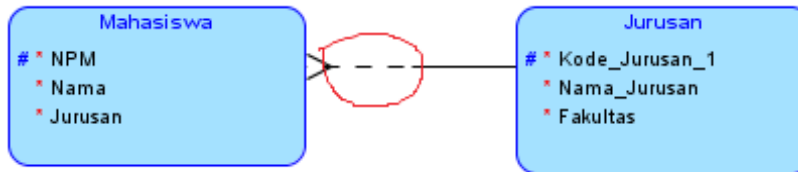
2. Relasi antar tabel Resep dan Obat



## b. Opsionalitas (Optionality)

Opsionalitas Relasi menunjukkan haruskah atau tidak harus adanya suatu relasi antara satu entitas dengan entitas lainnya. Opsionalitas ditandai dengan garis putus-putus.

Contoh :



Cara Membacanya : Setiap Mahasiswa harus memilih satu jurusan, sedangkan suatu jurusan tidak harus memiliki Mahasiswa.

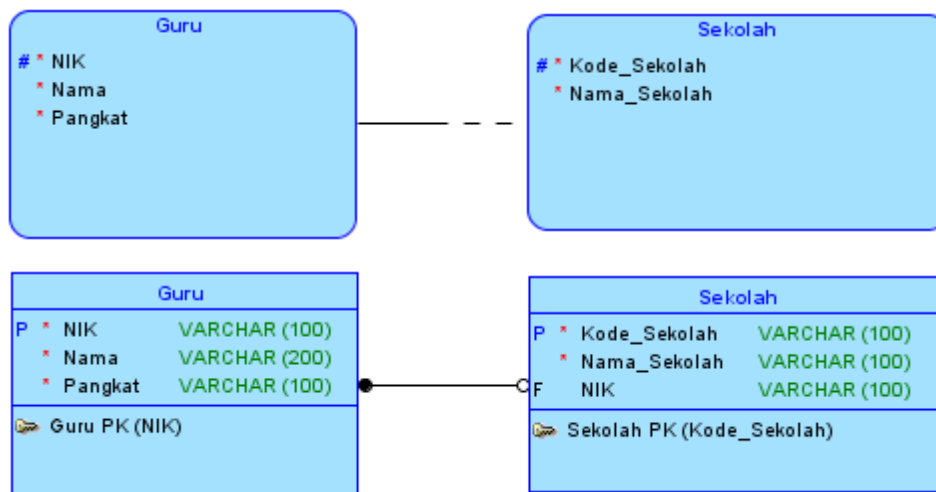
## C. Penggunaan Foreign Key

Foreign Key adalah atribut Primary key dari suatu tabel yang dipanggil di tabel lain yang memiliki relasi dengan tabel asal primary key tersebut. Foreign key digunakan sebagai identitas dari data yang akan diambil dari tabel yang saling berelasi.

Untuk setiap jenis relasi memiliki cara penggunaan Foreign Key yang berbeda :

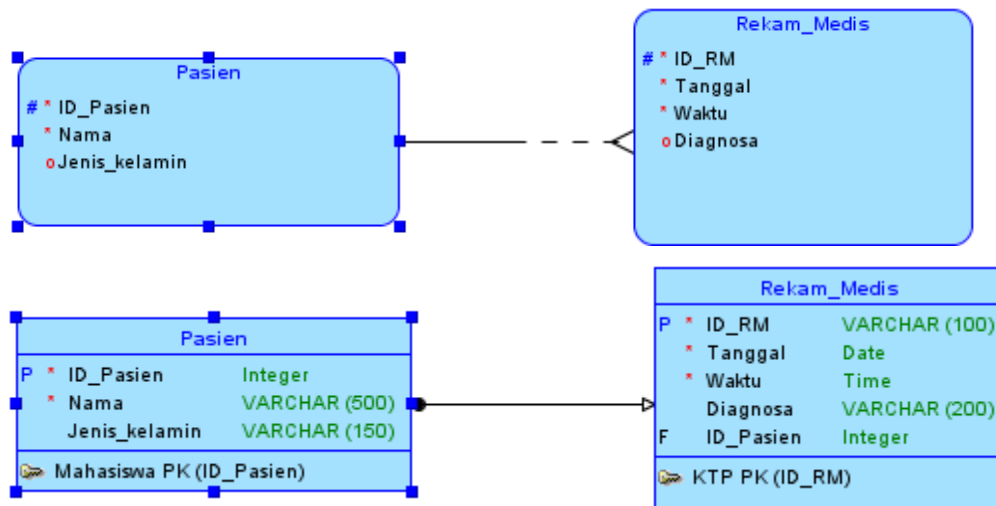
### 1) Relasi One to One

Pada relasi ini, Foreign key berada di tabel yang opsional.



Pada contoh diatas terdapat relasi Guru <Mengepalai> Sekolah, Foreign key berada di tabel Sekolah.

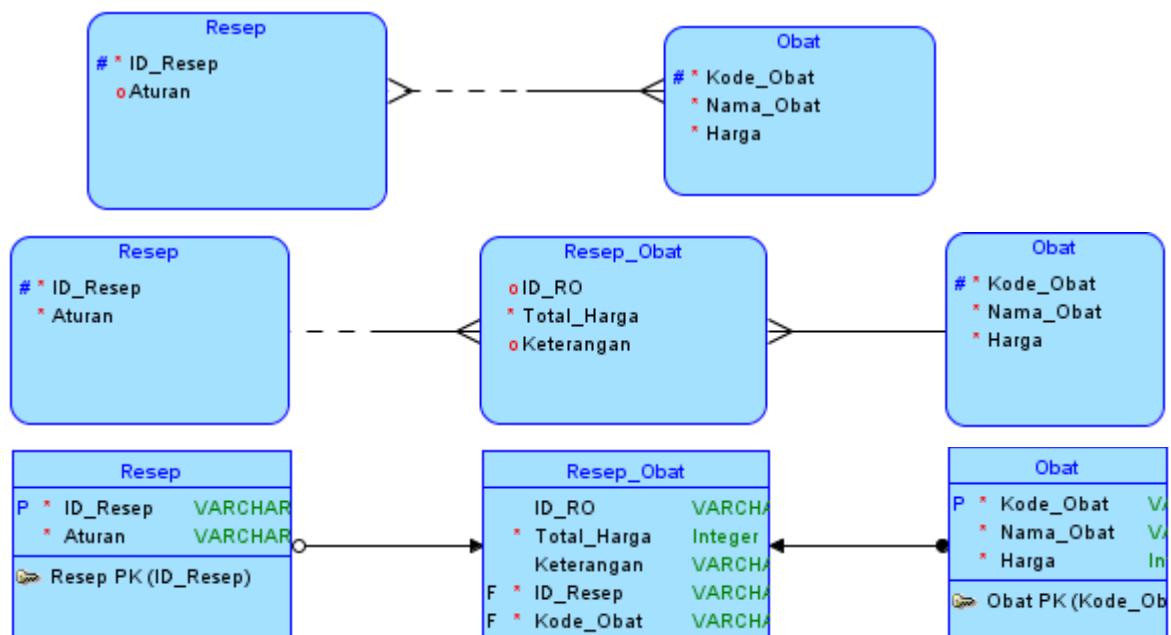
## 2) One to Many/ Many to One



Untuk Relasi ini Foreign key berada di tabel yang *Many* (crows foot).

## 3) Many to Many

Untuk Relasi jenis ini tidak bisa langsung direlasikan, sehingga perlu dibuat tabel bantu.



Maka kemudian Foreign Key nya kita letakkan pada tabel bantu tersebut.

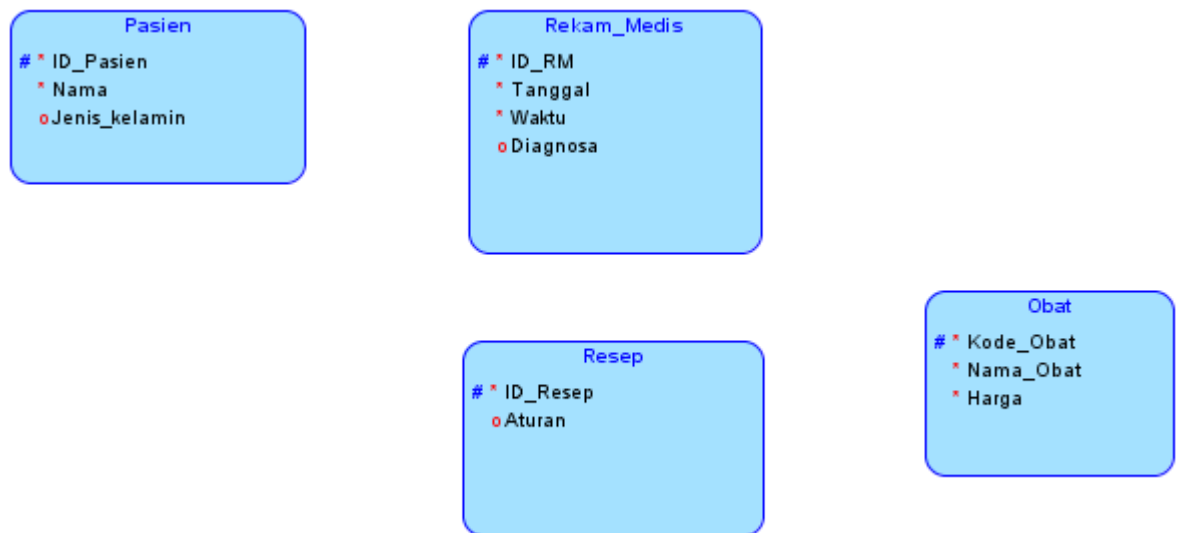
### Contoh Kasus 1:

Suatu Rumah Sakit ingin merancang database untuk menyimpan data rekam medis pasien. Database tersebut memiliki ketentuan sebagai berikut:

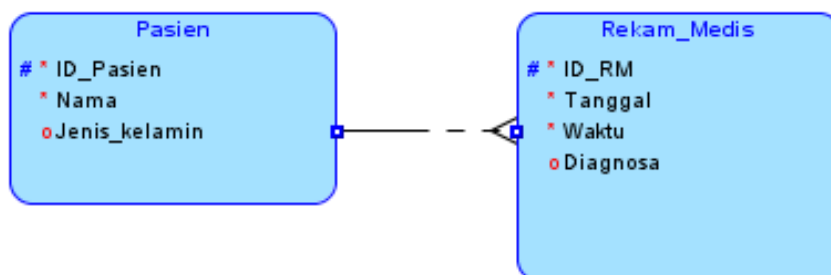
1. Satu rekam medis hanya boleh dimiliki oleh satu orang pasien, sedangkan satu orang pasien bisa memiliki lebih dari satu atau tidak memiliki rekam medis.
2. Satu rekam medis hanya boleh memuat satu resep obat, tetapi boleh juga tidak memuat resep obat.
3. Satu resep obat harus berisi minimal satu jenis obat.

### Penyelesaian :

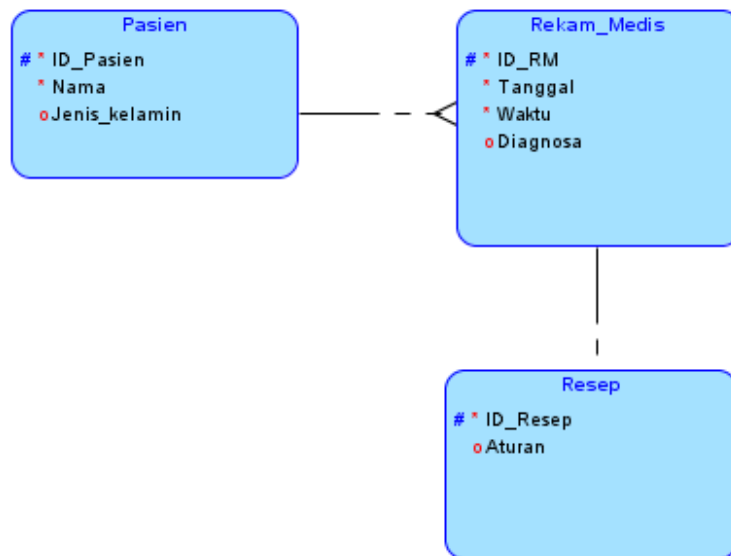
1. Kita perlu membuat 4 buah entity yaitu : Pasien, Rekam Medis, Resep, dan Obat



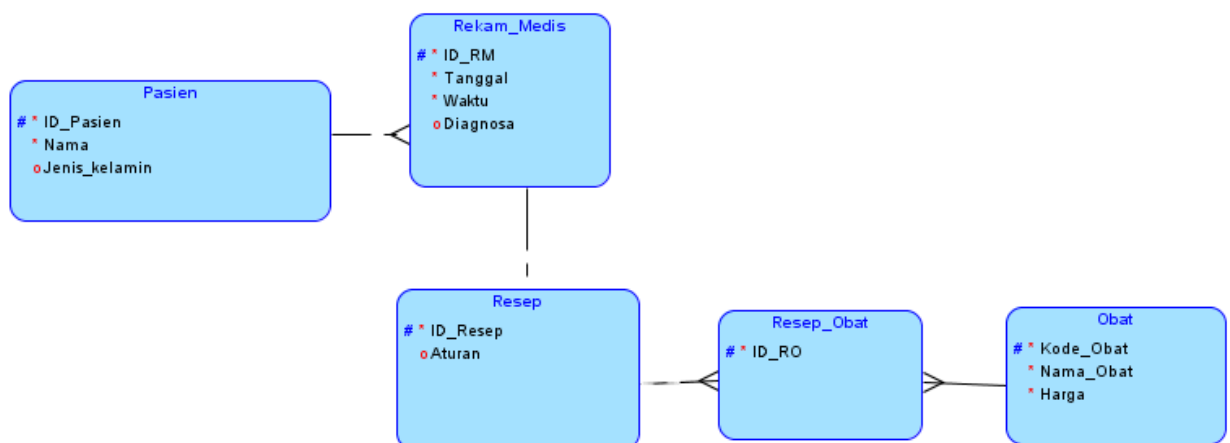
2. Dari ketentuan 1, maka relasi yang akan terbentuk adalah :



3. Dari ketentuan 2, maka relasi yang akan terbentuk adalah :



4. Untuk ketentuan 3 , kita tidak bisa langsung me-relasikan tabel Resep dan Obat, oleh karena itu kita perlu membuat tabel bantu.



## TUGAS :

### Contoh Kasus 2:

Suatu fakultas merancang database untuk menyimpan data mahasiswa dan dosen. Database tersebut memiliki ketentuan sebagai berikut:

1. Satu fakultas memiliki banyak jurusan, dan satu jurusan hanya dimiliki oleh 1 fakultas
2. Satu dosen hanya bisa mengepalai satu jurusan.
3. Satu dosen hanya bisa mengajar satu mata kuliah, sedangkan satu mata kuliah bisa diajar oleh banyak dosen.

Buatlah Conceptual Diagram dari kasus di atas.