

Transformasi Model E-R Ke Relasi

Setelah memahami berbagai istilah dalam model data relasional, kini saatnya untuk mempelajari cara mentransformasikan model E-R menjadi relasi.

1. Transformasi Tipe Entitas Kuat

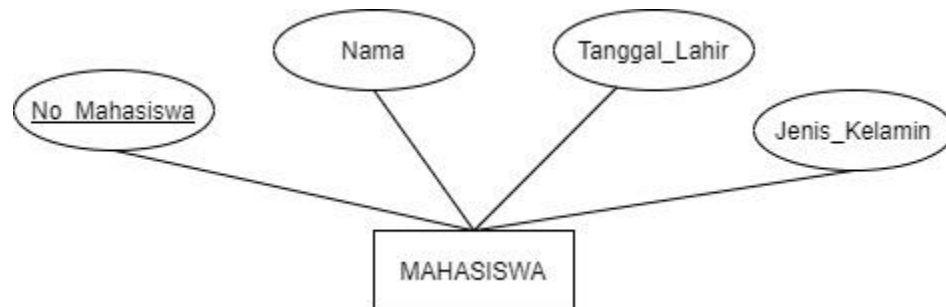
Suatu tipe entitas dibedakan menjadi tipe entitas kuat dan tipe entitas lemah. Tipe entitas kuat adalah tipe entitas yang keberadaannya tidak bergantung pada tipe entitas lain. Tipe entitas seperti itu dinyatakan dengan tanda kotak.

Ada perlakuan yang berbeda dalam melakukan transformasi tipe entitas kuat ke dalam relasi. Penentunya adalah jenis atribut, yakni berupa atribut sederhana, atribut komposit, atribut bernilai-ganda, ataupun atribut turunan.

a. Atribut Sederhana

Suatu atribut sederhana ditransformasikan ke dalam relasi menjadi atribut dalam relasi. Sebagai contoh, perhatikan gambar dibawah ini. Tipe entitas MAHASISWA mengandung tiga atribut sederhana bernama No_Mahasiswa, Nama, Tanggal_Lahir dan Jenis_Kelamin. Transformasi ke dalam relasi dilakukan dengan cara sebagai berikut:

1. Bentuk relasi dengan nama yang sama dengan nama dalam tipe entitas
2. Letakkan semua atribut dalam diagram E-R ke dalam relasi.
3. Bentuk kunci primer relasi tersebut berupa atribut yang menjadi kunci primer dalam tipe entitas.



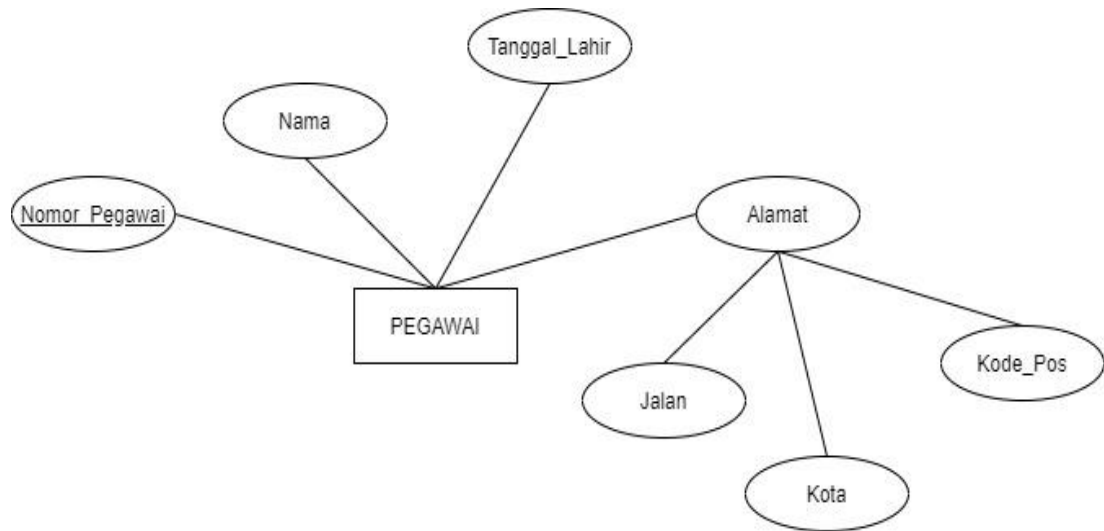
Relasi hasil transformasi dari diagram E-R adalah sebagai berikut:

MAHASISWA

No_Mahasiswa	Nama	Tanggal_Lahir	Jenis_Kelamin
--------------	------	---------------	---------------

b. Atribut Komposit

Jika terdapat atribut komposit, atribut tersebut tidak perlu dimasukkan ke dalam relasi, karena dapat mewakili oleh atribut sederhana yang menyusun atribut komposit tersebut. Contoh transformasi atribut komposit adalah sebagai berikut:



Relasi hasil transformasi dari diagram E-R adalah sebagai berikut:

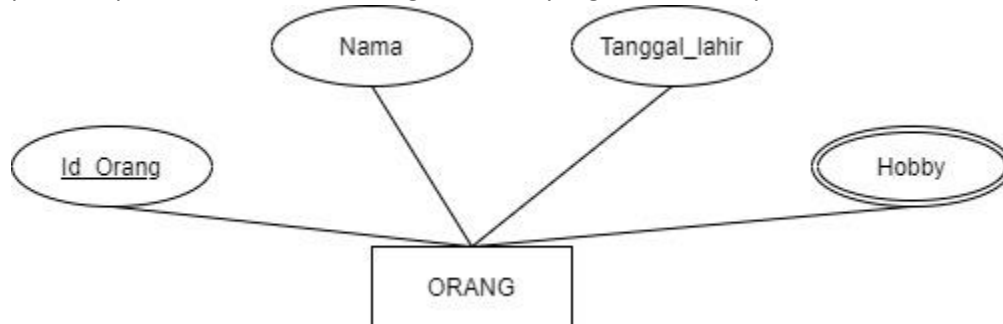
PEGAWAI

Nomor_Pegawai	Nama	Tanggal_Lahir	Jalan	Kota	Kode_Pos
---------------	------	---------------	-------	------	----------

Tampak bahwa Alamat tidak masuk dalam relasi PEGAWAI. Apabila terdapat kebutuhan untuk mendapatkan alamat, alamat dapat diperoleh melalui Jalan, Kota, Kode_Pos.

c. Atribut Bernilai Banyak

Jika terdapat atribut bernilai-banyak atribut tersebut akan membentuk relasi tambahan. Nama relasi tambahan ini dapat menggunakan nama yang mencerminkan atribut tersebut. Adapun kunci primer relasi tambahannya berupa kunci primer yang sesuai dengan kunci primer tipe entitas ditambah dengan atribut yang bernilai-banyak.



Relasi hasil transformasi dari diagram E-R adalah sebagai berikut:

ORANG

<u>Id Orang</u>	Nama	Tanggal_Lahir
-----------------	------	---------------

HOBBY

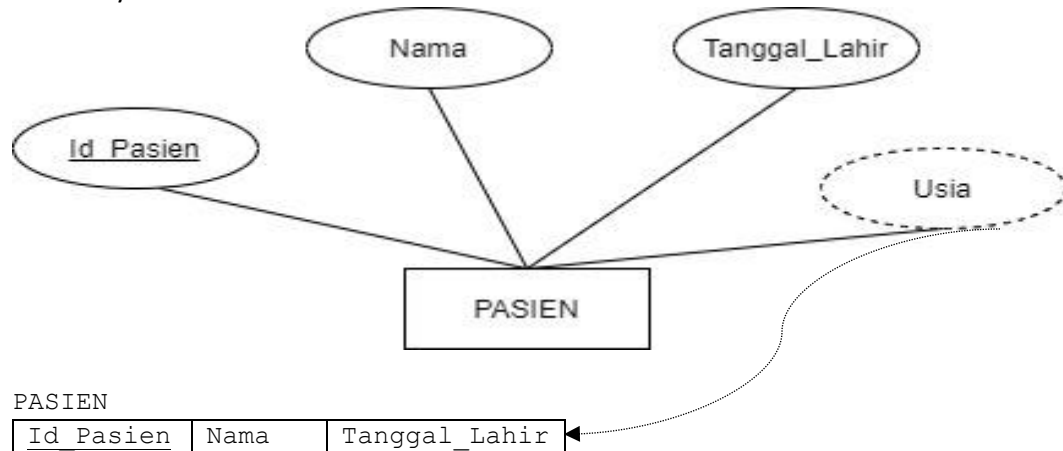
<u>Id Orang</u>	<u>Hobby</u>
-----------------	--------------

Tanda anak panah menunjukkan hubungan antarrelasi.

d. Atribut Turunan

Atribut Turunan adalah atribut yang dapat dihitung atau diturunkan dari nilai suatu atribut atau sejumlah atribut yang tersimpan dalam *database* atau dengan nilai lain (misalnya jam sistem atau tanggal sistem). Atribut seperti ini tidak perlu disertakan dalam relasi. Namun harus tetap dipastikan bahwa memang atribut tersebut dapat diperoleh melalui atribut lain atau melalui suatu perhitungan tertentu.

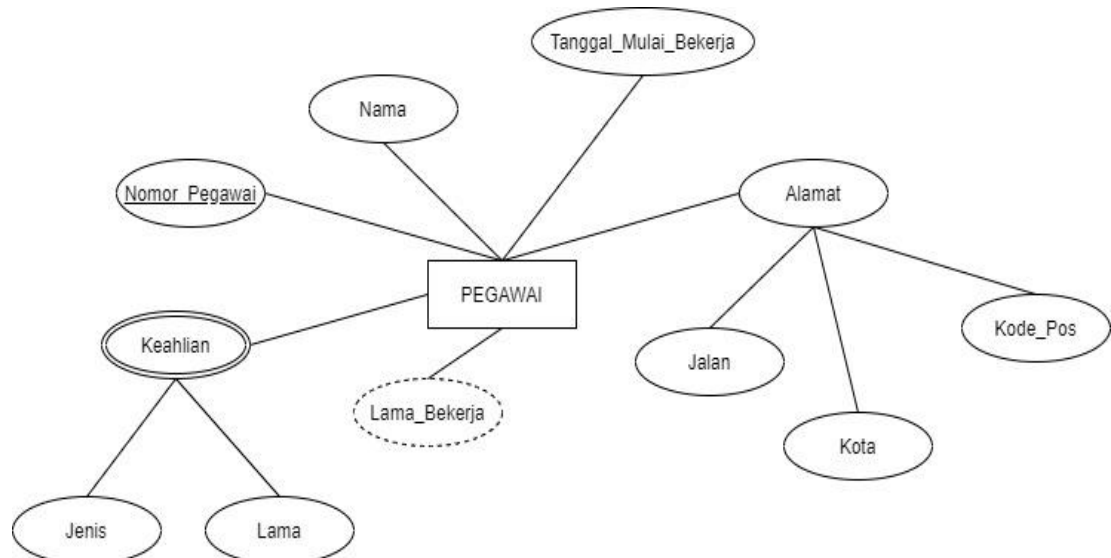
Contohnya:

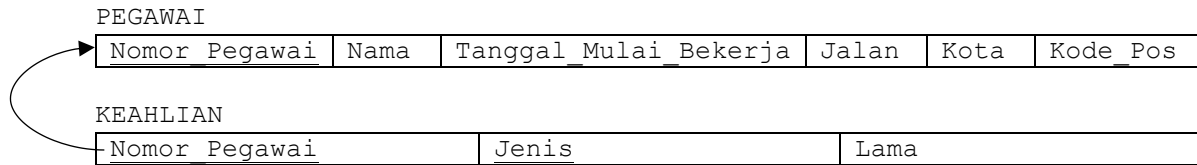


Tanda panah tersebut menunjukkan bahwa *Usia* tidak perlu masuk dalam relasi, karena dapat dihitung melalui *Tanggal_Lahir* dan tanggal sekarang pada sistem komputer.

e. Gabungan Berbagai Atribut

Pada praktik, dapat kita jumpai suatu entitas yang melibatkan berbagai jenis atribut. Secara prinsip, transformasi ke dalam relasi adalah dengan menggunakan prinsip-prinsip yang telah dibahas.

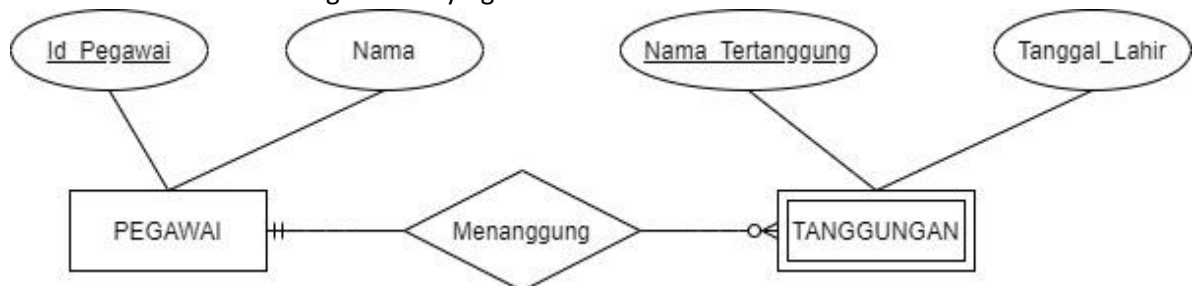




Tampak Bahwa:

1. Atribut Lama_Bekerja(karena dapat dihitung dari Tanggal_Mulai_Bekerja) tidak disertakan pada relasi PEGAWAI
 2. Atribut Alamat tidak disertakan karena kedudukannya sebagai atribut komposit. Yang dimasukkan dalam relasi PEGAWAI adalah atribut-atribut sederhana yang menyusun atribut komposit tersebut.
 3. Atribut keahlian tidak diletakkan pada relasi PEGAWAI mengingat berkedudukan sebagai atribut bernilai-ganda dan sebagai gantinya diletakkan pada relasi lain, yaitu relasi KEAHLIAN
 4. Atribut keahlian tidak diletakkan dalam relasi KEAHLIAN mengingat atribut ini berkedudukan sebagai atribut komposit. Yang dimasukkan dalam relasi KEAHLIAN adalah atribut-atribut sederhana yang menyusun atribut keahlian
 5. Relasi KEAHLIAN ditambah dengan atribut yang berasal dari kunci primer tipe entitas PEGAWAI. Kunci primer relasi KEAHLIAN adalah atribut tambahan ini dan juga salah satu atribut yang berasal dari atribut komposit keahlian. Dalam hal ini yang dipilih adalah Jenis mengingat atribut tersebut dapat menjadi pembeda dibandingkan dengan atribut Lama(yang menyatakan berapa lama si pegawai mempunyai jenis keahlian tersebut).
 6. Nomor_Pegawai pada relasi KEAHLIAN bertindak sebagai kunci asing yang merujuk ke kunci primer relasi PEGAWAI.
2. Transformasi Tipe Entitas Lemah

Tipe entitas lemah adalah tipe entitas yang keberadaannya bergantung pada tipe entitas lain. Tipe Entitas seperti ini tidak memiliki atribut pengenalan. Sebagai contoh, tipe entitas TANGGUNGAN berisi data anak para pegawai yang ditanggung oleh perusahaan (misalnya masalah Kesehatan). Keberadaan tipe entitas ini bergantung pada tipe entitas PEGAWAI. Dalam hal ini, tipe entitas PEGAWAI dinamakan sebagai identifying owner.

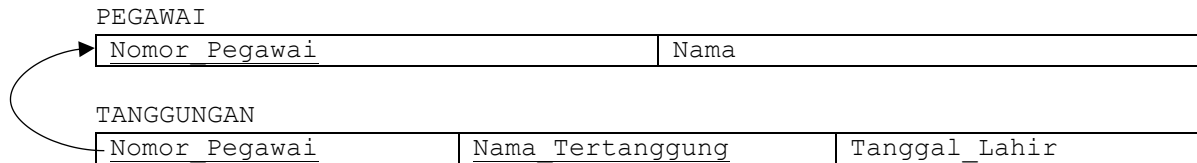


Atribut-atribut yang terkandung dalam TANGGUNGAN berupa Nama_Tertanggung dan Tanggal_Lahir. Tampak bahwa tidak ada atribut yang bertindak sebagai kunci primer. Tanda garis ganda pada Nama_Tertanggung menyatakan bahwa ia bertindak sebagai pengenalan parsial.

Transformasi tipe entitas lemah dilakukan dengan cara sebagai berikut:

1. Buat relasi yang namanya sesuai dengan nama pada tipe entitas
2. Masukkan kunci primer milik tipe entitas yang bertindak sebagai identifying owner ke relasi tersebut.

3. Pindahkan atribut-atribut tipe entitas lemah ke dalam relasi tersebut dengan mengikuti aturan seperti pada tipe entitas kuat.
4. Jadikan atribut yang berasal dari identifying owner dan pengenalan parsial sebagai kunci primer yang bersifat komposit.



Transformasi Hubungan Binary

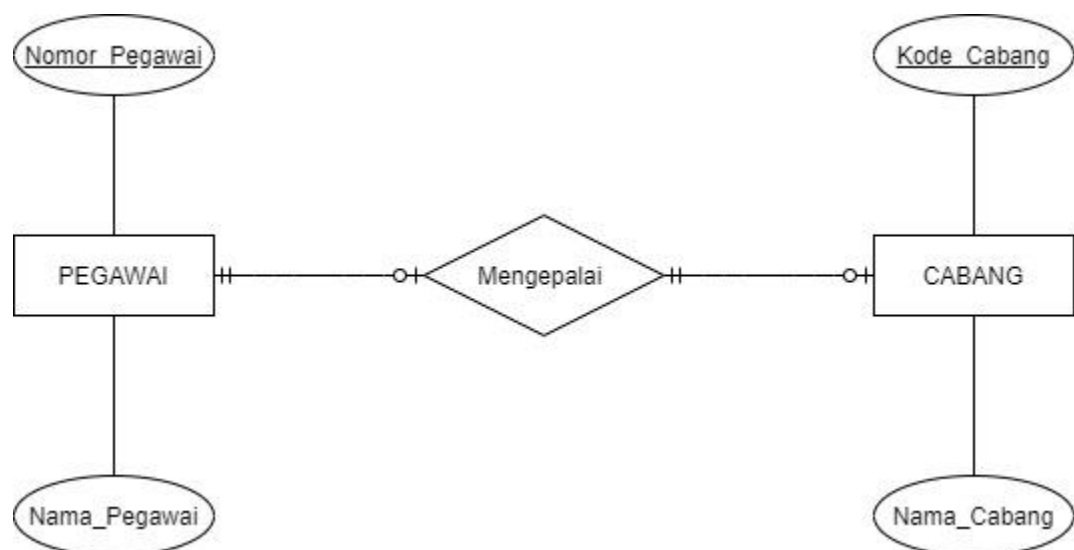
Hubungan binary adalah hubungan yang melibatkan dua tipe entitas. Hubungan yang terjadi dapat berupa 1:1, 1:M maupun M:N. Masing-masing jenis hubungan ini akan dibahas secara tersendiri.

1. Hubungan 1:1

Hubungan binary yang bersifat 1:1 ditransformasikan ke dalam bentuk relasi melalui prosedur tersebut.

- Transformasikan masing-masing tipe entitas ke dalam bentuk relasi, termasuk dengan kunci primernya. Aturan-aturan yang berlaku di depan bisa digunakan untuk menyusun pembentukan relasi.
- Tambahkan kunci primer salah satu relasi ke relasi yang lain untuk membentuk hubungan kunci asing (foreign key) dan kunci primer.

Sebagai contoh, Gambar menunjukkan hubungan binary yang bersifat 1:1 dan hasil petransformasian dalam bentuk relasi. Hubungan tersebut menunjukkan bahwa seorang pegawai menjadi kepala satu cabang atau tidak dan setiap kantor cabang hanya dipimpin oleh seorang pegawai.



PEGAWAI

<u>Nomor Pegawai</u>	Nama Pegawai
----------------------	--------------

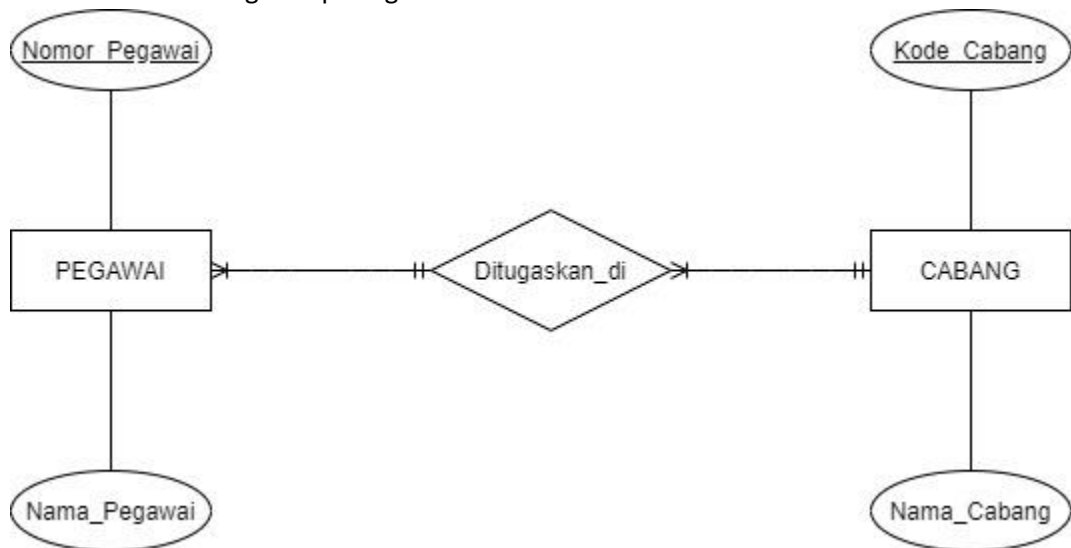
CABANG

<u>Kode Cabang</u>	Nama Cabang	<u>Nomor Pegawai</u>
--------------------	-------------	----------------------

Dalam hubungan one to one, tabel cabang juga memiliki foreign key. Kenapa ada atribut Nomor_Pegawai dalam tabel Cabang? Kenapa bukan di tabel Pegawai? Karena jika PEGAWAI ditambahkan dengan atribut yang berasal dari tabel CABANG, maka akan banyak baris yang KODE_CABANG-nya tidak terisi(atau diisikan dengan NULL, yang berarti bahwa atribut tersebut tidak berisi suatu nilai). Padahal dalam kasusnya sedikit pegawai yang akan mengepalai CABANG. Beda hal jika primary key PEGAWAI yang diberikan ke tabel CABANG, karena setiap cabang pasti memiliki seorang pegawai yang menjadi pimpinannya.

2. Hubungan 1:M

Pada hubungan binary yang bersifat 1:M, tipe entitas masing-masing menjadi satu relasi tersendiri dengan tambahan berupa penyertaan kunci primer tipe entitas yang berisi “satu” ke dalam relasi hasil transformasi tipe entitas yang berisi “banyak”. Contoh transformasi hubungan seperti gambar dibawah ini:



PEGAWAI

<u>Nomor Pegawai</u>	Nama Pegawai	<u>Kode Cabang</u>
----------------------	--------------	--------------------

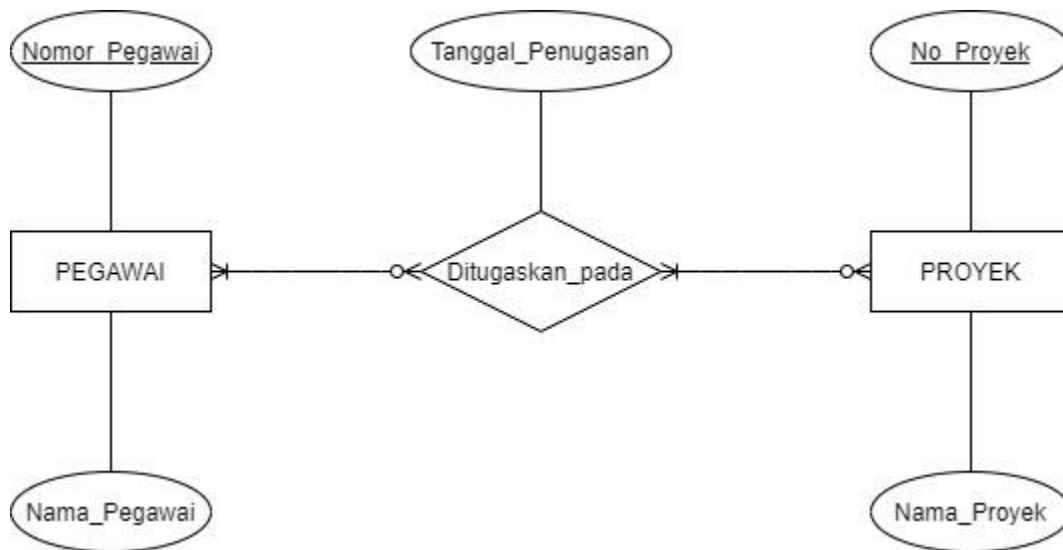
CABANG

<u>Kode Cabang</u>	Nama Cabang
--------------------	-------------

Pada tabel pegawai terdapat foreign key Kode_Cabang yang merujuk ke Kode_Cabang pada Table CABANG.

LANGKAH PRAKTIKUM

1. Perhatikan contoh gambar berikut merupakan contoh kasus hubungan many to many. Pada hubungan binary yang bersifat M:N, akan terbentuk relasi tambahan. Secara umum, prosedur untuk melakukan transformasi ke relasi adalah sebagai berikut:



2. Transformasikan ke dalam relasi tabel dengan membuat bentuk relasi-relasi yang sesuai dengan kedua tipe entitas dengan mengikuti aturan-aturan yang telah dibahas di depan. Kunci primer (Primary Key) kedua relasi adalah sama dengan atribut yang menjadi kunci primer (primary key) pada masing-masing tipe entitas.
3. Bentuk relasi baru yang mewakili hubungan Entitas.
4. Tambahkan ke dalam relasi, atribut yang merupakan Primary Key kedua tipe entitas (red:entitas).
5. Jadikan kedua atribut tersebut sebagai kunci komposit.
6. Pindahkan atribut-atribut dalam hubungan ke relasi.

7. Hasilnya ada tiga tabel yaitu PEGAWAI, PROYEK, PEGAWAI_PROYEK

PEGAWAI

<u>Nomor_Pegawai</u>	Nama_Pegawai
----------------------	--------------

PROYEK

<u>No_Projek</u>	Nama_Projek
------------------	-------------

PEGAWAI_PROYEK

<u>Nomor_Pegawai</u>	<u>No_Projek</u>	Tanggal_Penugasan
----------------------	------------------	-------------------

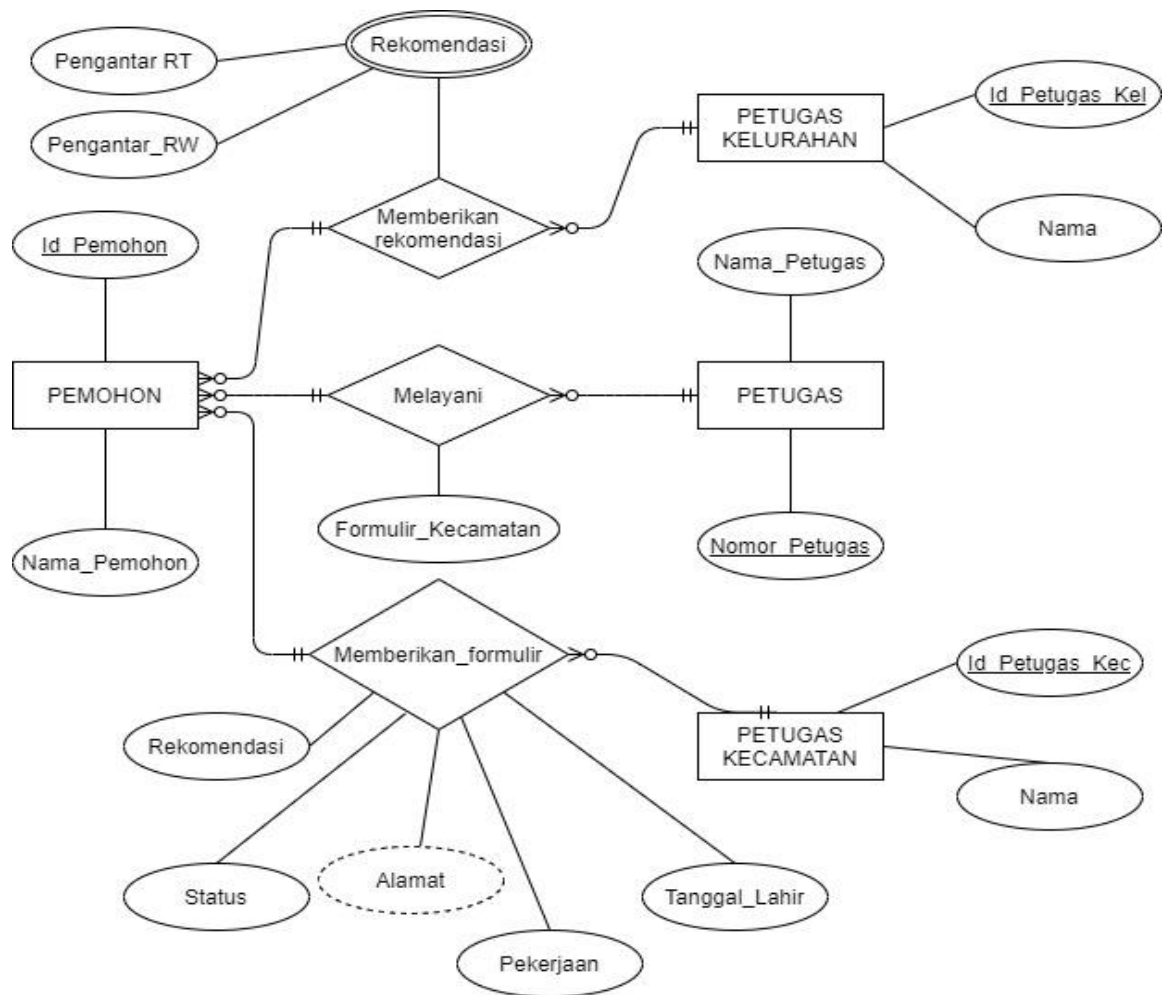
8. Tunjukkan dengan tanda panah, hubungan relasinya. Seperti contoh (dasar toeri) yang ada diatas.

Contoh Kasus:

Ketika seseorang akan membuat E-KTP secara online melalui pelayanan Disdukcapil, pertama seseorang tersebut harus memiliki pengantar RT/RW untuk dibawa ke kelurahan. Setelah dari kelurahan membawa formulir E-KTP yang sudah diisikan di Kecamatan. Kemudian jika berkas sudah lengkap, akan diproses dalam pelayanan Disdukcapil. Pada Sistem Informasi Manajemen (SIM) Disdukcapil setempat, petugas akan memasukkan data e-KTP dan kemudian mencetaknya.

Analisa kasus dengan mengidentifikasi entitas kemudian ditransformasikan kedalam tabel relasi:

1. Ada 4 aktor yaitu pemohon KTP, kelurahan, kecamatan dan petugas Disdukcapil sehingga kita mempunyai 4 entitas.
2. Pada realisasinya, petugas disdukcapil rasionya lebih sedikit dibandingkan dengan pemohon. Meskipun ada suatu keadaan seperti tidak adanya pemohon dalam hari tertentu. Hal tersebut dapat dianalisa bahwa petugas dapat tidak menginputkan data seorang pemohon sekalipun. Begitu juga untuk hubungan Petugas Kelurahan ke pemohon dan Petugas Kecamatan ke pemohon.
3. Dari poin dua, kita menganalisanya hubungan setiap entitas adalah one to many setiap hubungannya.
4. Jika digambarkan dalam sebuah entitas adalah sebagai berikut:



5. Diketahui pemohon harus meminta surat rekomendasi ke RT/RW setempat sehingga terdapat atribut bernilai-banyak yaitu Rekomendasi yang bernilai apakah dia sudah memiliki rekomendasi dari RT/RW. Jika sudah proses akan dilanjutkan ke Kecamatan. PETUGAS KECAMATAN akan memberikan formulir jika sudah mendapatkan rekomendasi dari Kelurahan. Atribut alamat dapat dibuat sebagai atribut turunan karena dari awal kita meminta rekomendasi sudah diketahui alamat si pemohon. Untuk kemudian petugas disdukcapil dapat memproses pencetakan KTP berdasarkan formulir kecamatan.
6. Analisa dalam Model E-R merupakan representasi asumsi kita terhadap kasus tersebut sehingga untuk penilaian kebenarannya hanya tepat dan tidak tepat.
7. Buatlah tabel relasinya hanya hubungan PETUGAS ke PEMOHON, PETUGASKELURAHAN KE PEMOHON dan PETUGAS KECAMATAN KE PEMOHON.