



Perancangan Basis Data Menggunakan ER Model

Finsa Nurpandi, M.T.
Universitas Suryakencana
2021



The Topics

1. **Overview of the Design Process**
2. **The Entity-Relationship Model**
3. **Complex Attributes**
4. **Mapping Cardinalities**
5. **Weak Entity Set**
6. **Removing Redundant Attributes in Entity Sets**
7. Reducing E-R Diagrams to Relational Schemas
8. Extended E-R Features
9. E-R Design Issues
10. Alternative Notations for Modeling Data



1. Overview of the Design Process



- Tugas membuat aplikasi basis data adalah tugas yang kompleks, melibatkan perancangan dari skema basis data, rancangan dari program yang mengakses dan memperbaharui data, dan merancang skema keamanan untuk mengontrol akses ke data
- Kebutuhan pengguna memainkan peranan penting dari proses perancangan.



1.1 Fase Perancangan

- Perancang basis data harus berinteraksi dengan pengguna aplikasi untuk memahami kebutuhan dari aplikasi, mewakili aplikasi dengan cara tingkat tinggi yang dapat dipahami oleh pengguna, dan menterjemahkan kebutuhan kedalam perancangan tingkat rendah.
- Model data tingkat tinggi melayani perancang basis data dengan menyediakan kerangka kerja konseptual untuk menentukan, secara sistematis, persyaratan data pengguna basis data, dan struktur basis data yang memenuhi kebutuhannya.
- Selanjutnya merupakan tahapan perancangan basis data.



1. Mengkarakteristikan sepenuhnya kebutuhan data calon pengguna basis data. Perancang basis data perlu berinteraksi secara ekstensif dengan pengguna. Hasil dari tahapan ini adalah spesifikasi kebutuhan pengguna.
2. Memilih model data. Menerjemahkan kebutuhan menjadi skema konseptual dari basis data. Biasanya, fase konseptual disain menghasilkan pembuatan diagram hubungan entitas yang menyediakan representasi grafik dari skema tersebut. Perancang meninjau skema untuk memastikan bahwa semua kebutuhan data memenuhi dan tidak bertentangan dengan yang lainnya.



3. Skema konseptual yang dikembangkan sepenuhnya menunjukkan kebutuhan dari perusahaan. Dalam spesifikasi kebutuhan fungsional, pengguna menggambarkan jenis operasi (atau transaksi) yang akan dilakukan pada data.
4. Proses perpindahan dari model data abstrak ke implementasi basis data berlangsung dalam dua tahap:
 - a. **Tahap desain logis.** Perancang memetakan skema konseptual tingkat tinggi ke dalam model data implementasi sistem basis data. Model data implementasi biasanya adalah model data relasional. Biasanya didefinisikan menggunakan model hubungan entitas ke dalam skema relasi.
 - b. **Tahap desain fisik.** Perancang menggunakan hasil dari penentuan skema sistem basis data, dimana fitur fisik basis data ditentukan. Fitur tersebut terdiri dari format organisasi file dan pemilihan struktur indeks.



1.2 Alternatif Perancangan

- Bagian utama dari proses desain basis data adalah memutuskan bagaimana merepresentasikan berbagai jenis “benda” seperti orang, tempat, produk, dan sejenisnya. “Benda” tersebut, dijelaskan dengan istilah **entitas**.
- Berbagai entitas terkait satu sama lainnya dalam berbagai cara, yang semuanya perlu ditampilkan dalam rancangan basis data.



- Dalam merancang skema basis data, terdapat dua hal yang harus dihindari:
1. **Redundansi.** Redundansi dapat terjadi dalam skema relasional. Masalah terbesar dari menampilkan informasi yang redundan adalah bahwa sebuah informasi dapat menjadi tidak konsisten jika informasi tersebut diperbaharui tanpa mengambil tindakan pencegahan untuk memperbaharui semua Salinan informasi.
 2. **Ketidaklengkapan.** Rancangan yang buruk dapat membuat aspek-aspek tertentu menjadi tidak dapat dimodelkan. Misalkan entitas memiliki hubungan tunggal dimana mengulang semua informasi hanya sekali. Sehingga tidak bisa merepresentasikan informasi baru. Solusi diantaranya dengan menyimpan null pada bagian informasi. Solusi tersebut dapat dicegah juga dengan constraint primary-key.



2. The E-R Model



- Entity-relationship (E-R) data model digunakan sebagai fasilitas perancangan basis data dengan memperbolehkan spesifikasi skema perusahaan untuk menggambarkan keseluruhan struktur logis dari basis data.
- E-R model sangat berguna untuk memetakan arti dan interaksi dari perusahaan secara nyata menjadi skema konseptual.
- Model data E-R menggunakan tiga konsep dasar:
 - Himpunan entitas (*entity sets*)
 - Himpunan relasi (*relationship sets*)
 - Atribut (*attributes*)



2.1 Entity Sets

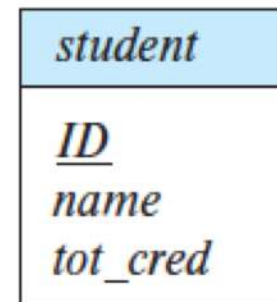
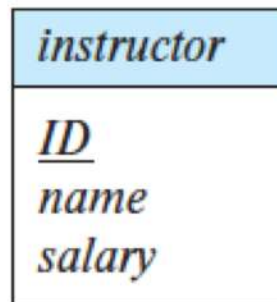
- Entitas (*entity*) merupakan “benda” atau “objek” di dunia nyata yang dapat dibedakan dari semua objek lainnya.
- Entitas memiliki sekumpulan property, dan nilai untuk beberapa kumpulan property harus mengidentifikasi entitas secara unik.
- Misalnya, setiap orang di Universitas adalah entitas. Setiap orang memiliki property id yang nilainya secara unik mengidentifikasi orang tersebut.



- Himpunan entitas adalah sekumpulan entitas dengan tipe yang sama yang memiliki property atau atribut yang sama.
- Sebagai contoh: himpunan entitas mahasiswa dapat mewakili himpunan semua mahasiswa di Universitas.
- Entitas diwakili oleh sekumpulan atribut. Atribut adalah sifat deskriptif yang dimiliki oleh setiap anggota himpunan entitas.
- Penunjukan atribut untuk himpunan entitas menyatakan bahwa basis data menyimpan informasi serupa mengenai setiap entitas dalam himpunan entitas. Namun, setiap entitas mungkin memiliki nilainya sendiri untuk setiap atribut.
- Setiap entitas memiliki nilai untuk setiap atributnya. Sebagai contoh entitas *instructor* mungkin memiliki nilai 12121 untuk atribut ID.



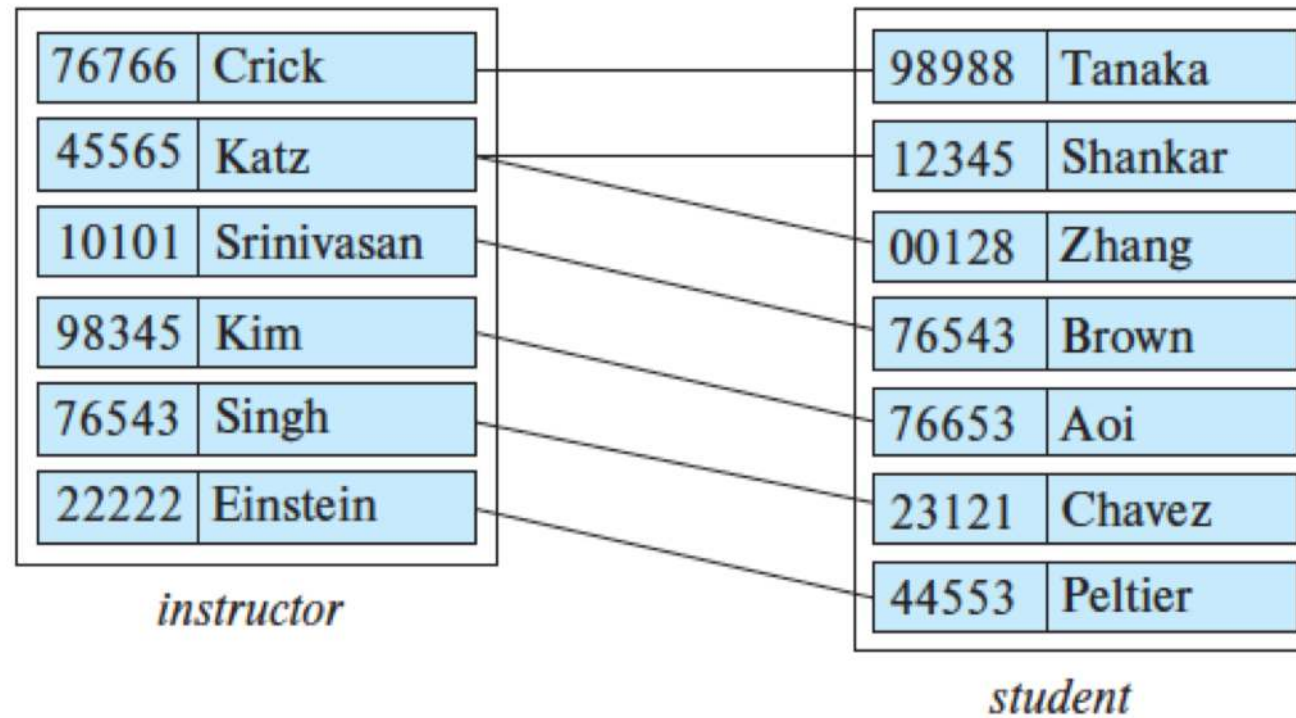
- Himpunan entitas digambarkan dalam E-R diagram dengan bentuk **empat persegi Panjang**, yang dibagi menjadi dua bagian.
- Bagian pertama, berisi nama dari himpunan entitas.
- Bagian kedua, berisi nama dari semua atribut dari himpunan entitas.





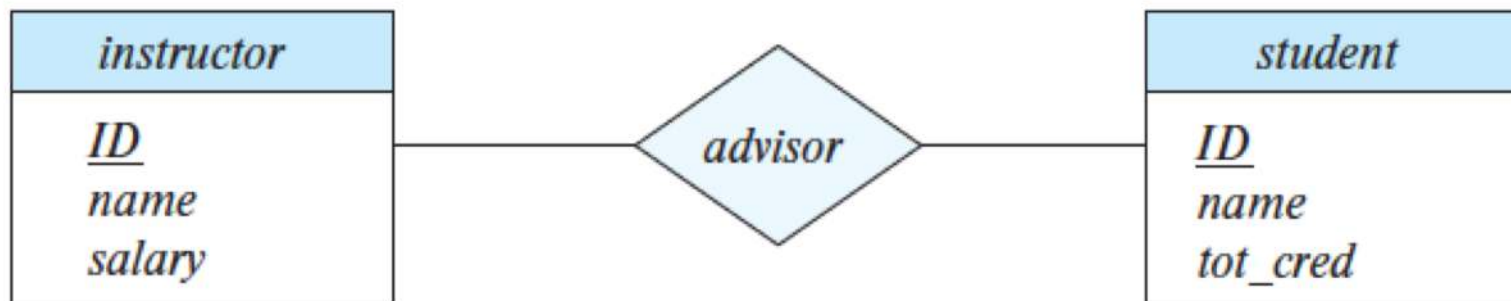
2.2 Relationship Sets

- Relasi adalah gabungan antara beberapa entitas. Himpunan relasi merupakan himpunan dari relasi yang memiliki tipe yang sama.
- Sebagai contoh, menentukan hubungan *advisor* yang menghubungkan instructor Katz dengan mahasiswa Shankar. Relasi ini menentukan bahwa Kats merupakan dosen wali dari Shankar.





- Himpunan relasi ditampilkan dalam diagram E-R dengan tanda **diamond**, dimana terhubung melalui garis ke beberapa himpunan entitas berbeda.
- E-R Diagram di bawah ini menunjukkan dua himpunan entitas *instructor* dan *student*, terkait melalui hubungan binery yang disebut *advisor*.

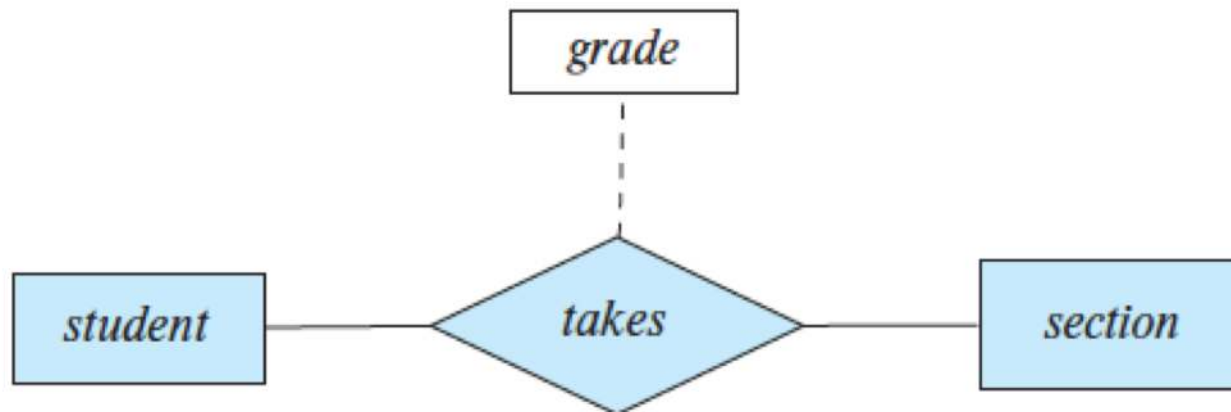




- Relasi mungkin akan memiliki atribut yang disebut dengan descriptive attributes.
- Sebagai contoh, misalkan himpunan relasi *takes* yang berhubungan dengan himpunan entitas *student* dan *section*.
- Kita dapat menyimpan descriptive attribute *grade* dengan relasi untuk mencatat grade dimana mahasiswa mendapatkan nilai dari kursus yang ditawarkan.



- Sebuah atribut pada himpunan relasi direpresentasikan pada E-R diagram dengan sebuah **undivided rectangle**.
- Menghubungkan *rectangle* dengan garis putus-putus pada diamond menggambarkan himpunan relasi.
- Seperti pada gambar di bawah ini, himpunan relasi takes diantara himpunan entitas *section* dan *student*.





- Amati bahwa atribut dari dua himpunan entitas telah dihilangkan dari E-R Diagram, dengan pemahaman bahwa atribut tersebut ditentukan di tempat lain dalam E-R Diagram lengkap untuk *university*.
- Desain E-R yang kompleks mungkin perlu dipecah menjadi beberapa diagram yang mungkin terletak di halaman yang berbeda
- Himpunan relasi harus ditampilkan hanya dalam satu lokasi, tetapi himpunan entitas dapat diulang dilebih dari satu lokasi.
- Atribut dari himpunan entitas harus ditampilkan pada kemunculan pertama. Kemunculan berikutnya dari himpunan entitas harus ditunjukkan tanpa atribut untuk menghindari pengulangan informasi dan kemungkinan inkonsistensi atribut.



3. Complex Attributes



- Untuk setiap atribut, terdapat himpunan nilai yang diperbolehkan, disebut dengan **domain** atau **value set**.
- Domain dari atribut *course_id* mungkin terdiri dari himpunan untai teks dari beberapa karakter.
- Begitu juga dengan domain dari atribut *semester* yang dapat berisi untai dari himpunan {Fall, Winter, Spring, Summer}



Setiap atribut yang digunakan pada model E-R, dapat dikategorikan berdasarkan pada tipe atribut.

- Atribut Simple dan composite.
- Atribut Single-valued dan Multivalued
- Atribut Derived



Atribut Simple dan Composite

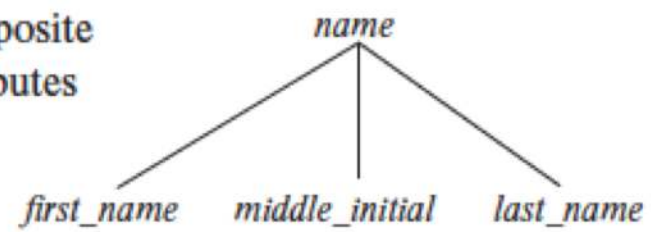
- Pada contoh-contoh sebelumnya, atribut yang digunakan adalah Simple, karena tidak dipecah menjadi beberapa sub bagian.
- Atribut composite, dapat dibagi menjadi beberapa sub bagian.
- Sebagai contoh, atribut *name* dapat distrukturkan menjadi atribut composite yang terdiri dari *first_name*, *middle_name*, dan *last_name*.



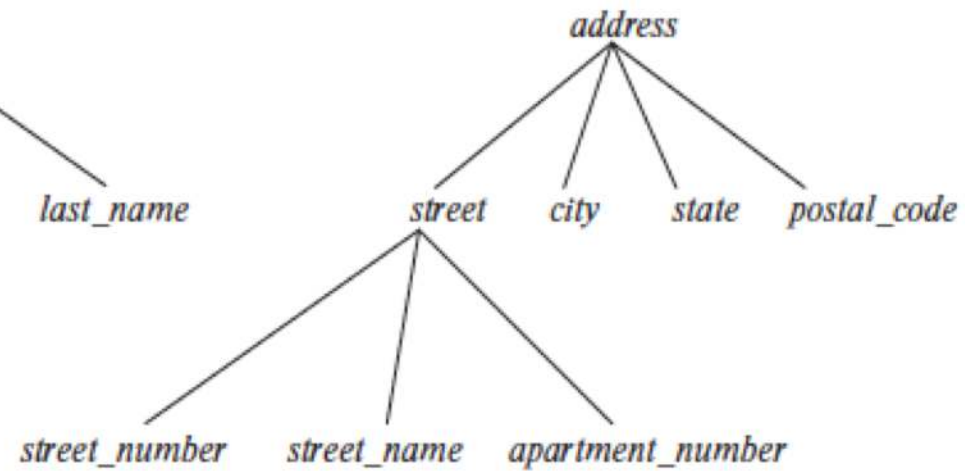
- Penggunaan atribut composite sangat berguna untuk:
 1. Pengguna ingin merujuk ke seluruh atribut pada beberapa kesempatan, dan hanya komponen atribut pada kesempatan lain.
 2. Membantu untuk mengelompokkan atribut yang berhubungan
 3. Membuat model menjadi lebih “cleaner”



composite
attributes



component
attributes





- Gambar selanjutnya adalah atribut composite yang direpresentasikan dalam notasi E-R.
- Notasi atribut multivalued dilambangkan dengan "{phone_number}"
- Notasi atribut derived dilambangkan dengan "age()"



instructor

ID

name

first_name

middle_initial

last_name

address

street

street_number

street_name

apt_number

city

state

zip

[phone_number]

date_of_birth

age ()



Atribut Single-valued dan Multivalued

- Pada contoh-contoh sebelumnya, atribut menggunakan single value pada beberapa entitas. Sebagai contoh, atribut `student_ID` untuk entitas mahasiswa tertentu hanya mengacu pada satu ID mahasiswa saja.
- Terdapat contoh dimana atribut memiliki satu set nilai untuk entitas tertentu. Misalkan menambahkan entitas instructor, atribut nomor telepon. Dimana seorang instructor mungkin dapat memiliki nol, satu, atau beberapa nomor telepon.



Atribut Derived

- Nilai untuk tipe atribut ini dapat berasal dari nilai atribut atau entitas yang terkait.
- Sebagai contoh, himpunan entitas *instructor* memiliki atribut `student_advised`, dimana merepresentasikan berapa banyak mahasiswa wali dari instructor tersebut.
- Nilai dari atribut `student_advised` diperoleh dengan menghitung jumlah entitas siswa yang terkait dengan instructor tersebut.



4. Mapping Cardinalities



- Mapping cardinalities atau cardinality ratios, menyatakan jumlah entitas lain dapat dikaitkan melalui himpunan relasi.
- Pemetaan kardinalitas sangat berguna dalam mendeskripsikan himpunan relasi biner, meskipun kardinalitas tersebut dapat berkontribusi pada deskripsi himpunan relasi yang melibatkan lebih dari dua himpunan entitas.

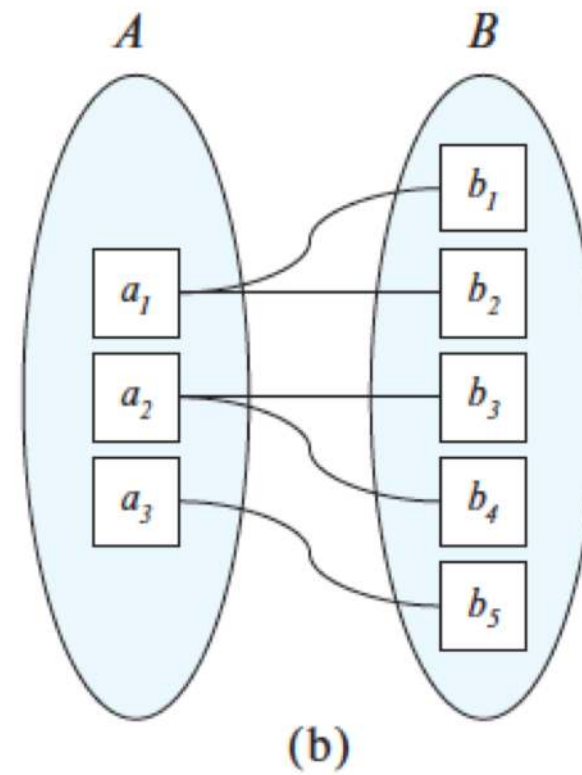
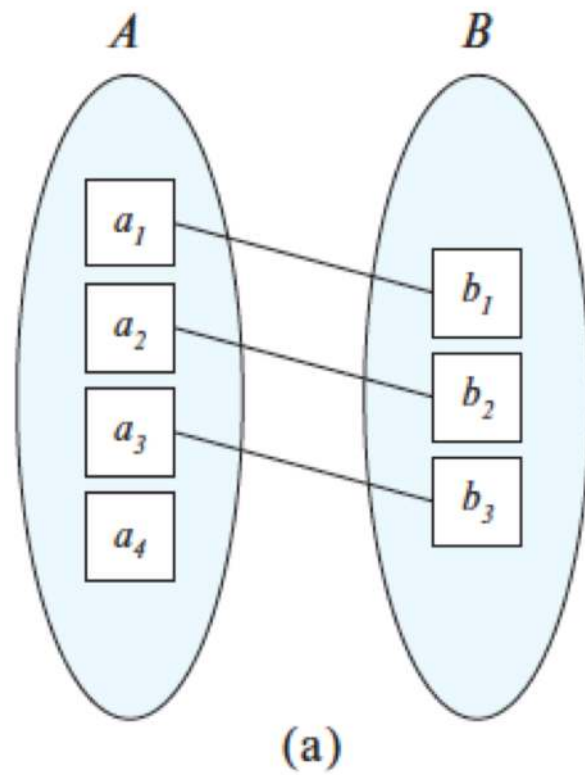


Untuk himpunan relasi biner R antara himpunan entitas A dan B , pemetaan kardinalitas harus dari salah satu berikut ini:

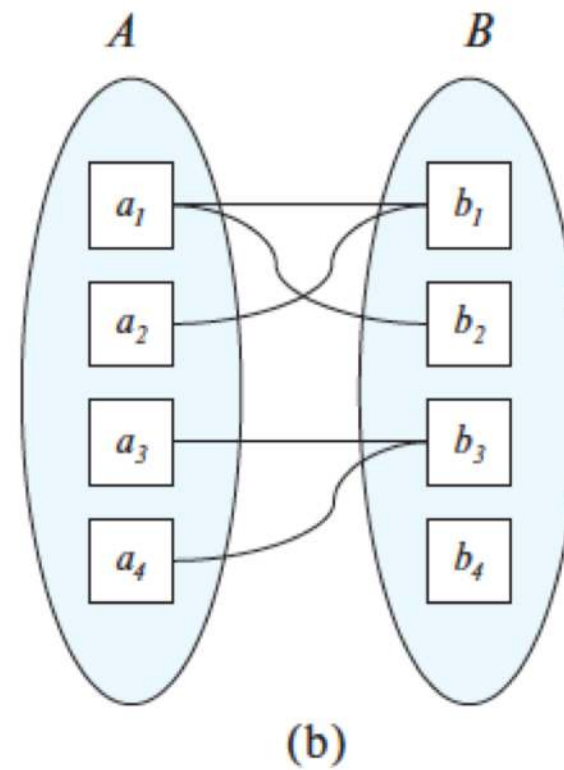
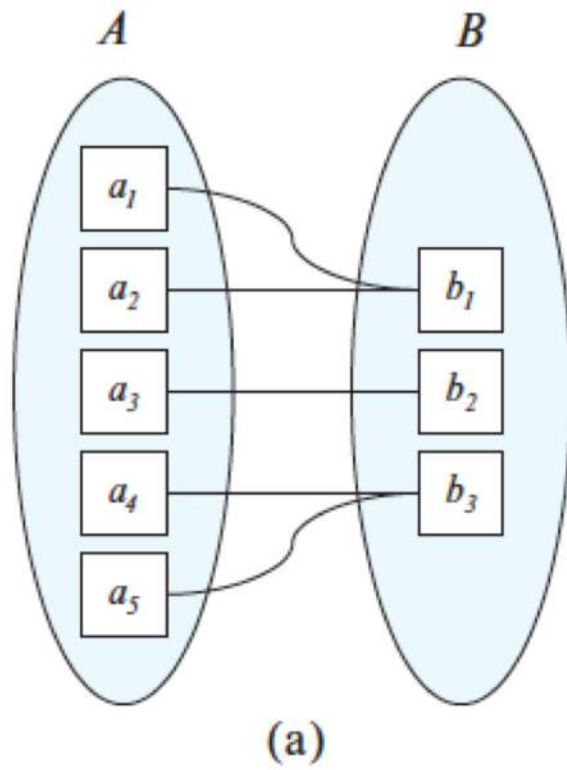
1. **One-to-one**. Suatu entitas A diasosiasikan dengan *paling banyak* satu entitas pada B , dan entitas B diasosiasikan dengan *paling banyak* satu entitas pada A .
2. **One-to-many**. Suatu entitas A diasosiasikan dengan jumlah berapapun (nol atau lebih) dari entitas B . sesuatu entitas B , dapat diasosiasikan dengan paling banyak satu entitas pada A .



3. **Many-to-one.** Suatu entitas dalam A diasosiasikan dengan paling banyak satu entitas pada B. suatu entitas pada B, dapat diasosiasikan dengan jumlah berapapun dari entitas A.
4. **Many-to-many.** Suatu entitas pada A diasosiasikan dengan jumlah berapa pun dari entitas pada B, dan entitas B diasosiasikan dengan jumlah berapapun dari entitas A.



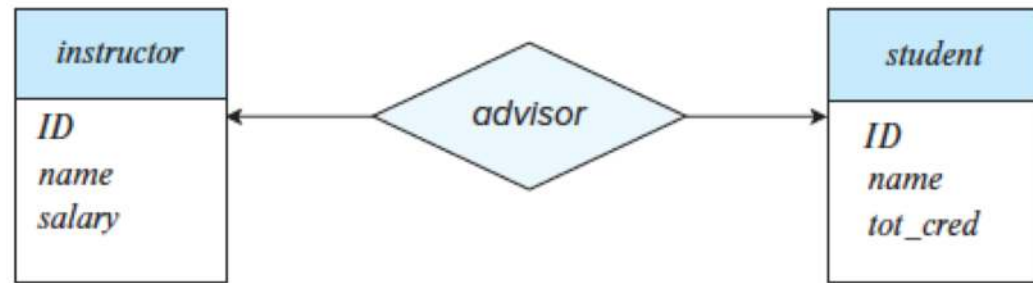
- (a) One-to-one
(b) One-to-many



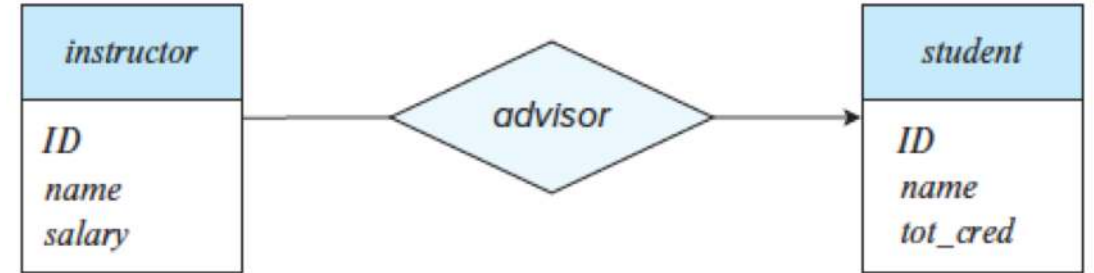
- (a) Many-to-one
- (b) Many-to-many



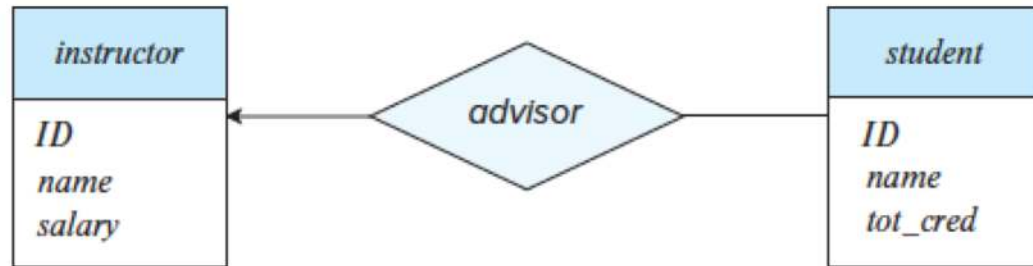
- Pada notasi E-R Diagram, kardinalitas pada suatu hubungan ditunjukkan dengan menggambar garis panah (\rightarrow) atau garis tanpa panah (-) diantara himpunan relasi dan himpunan entitas.
- Garis panah untuk menggambarkan *one*
- Garis tanpa panah menggambarkan *many*



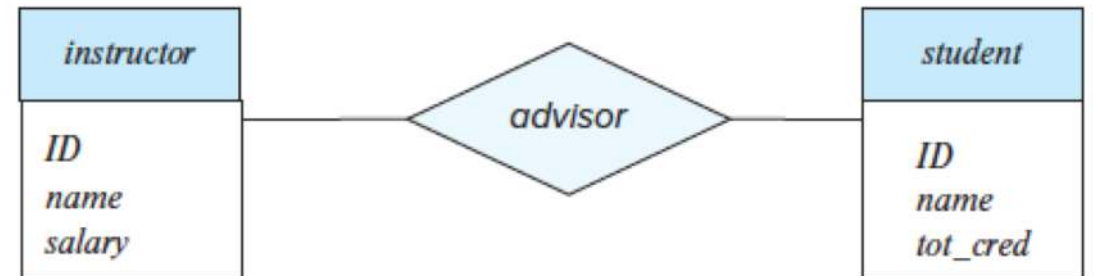
(a) One-to-one



(c) Many-to-one



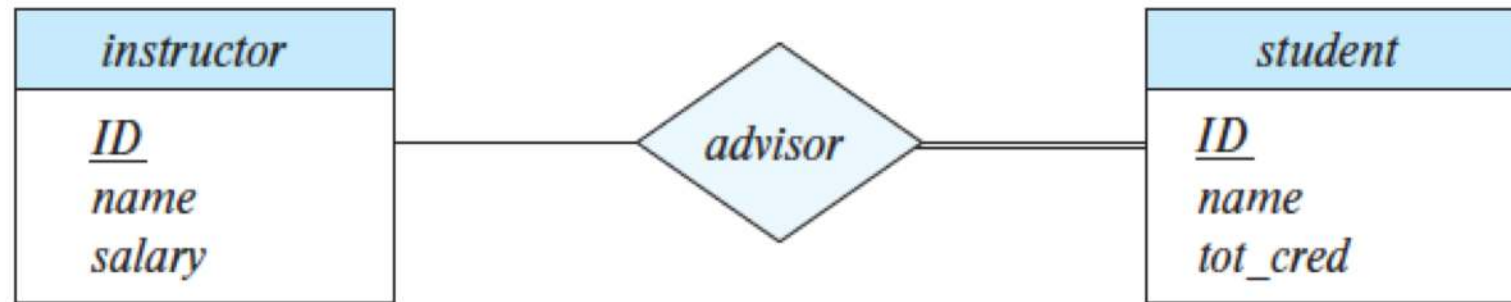
(b) One-to-many



(d) Many-to-many



- Partisipasi himpunan entitas E dalam himpunan relasi R dikatakan **Total** jika setiap entitas pada E harus berpartisipasi paling sedikit satu relasi di R .
- Jika ada kemungkinan beberapa entitas pada E tidak berpartisipasi dalam relasi R , partisipasi himpunan entitas E dalam R dikatakan **Parsial**.

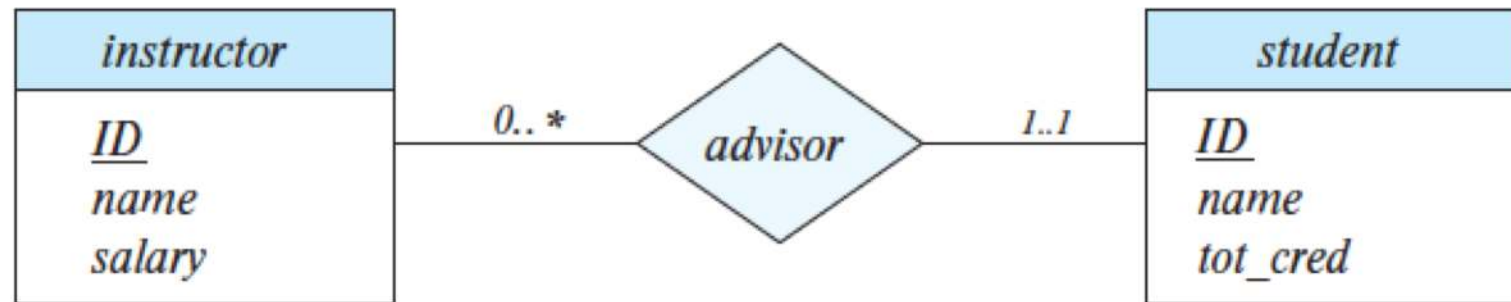




- E-R diagram menyediakan cara lain untuk menunjukan kendala yang lebih kompleks pada setiap entitas yang berpartisipasi dalam relasi dalam suatu himpunan relasi.
- Sebuah garis memiliki asosiasi kardinalitas minimum dan maksimum, seperti dalam bentuk $l...h$.
- Dimana l merupakan minimum, dan h merupakan kardinalitas maksimum.

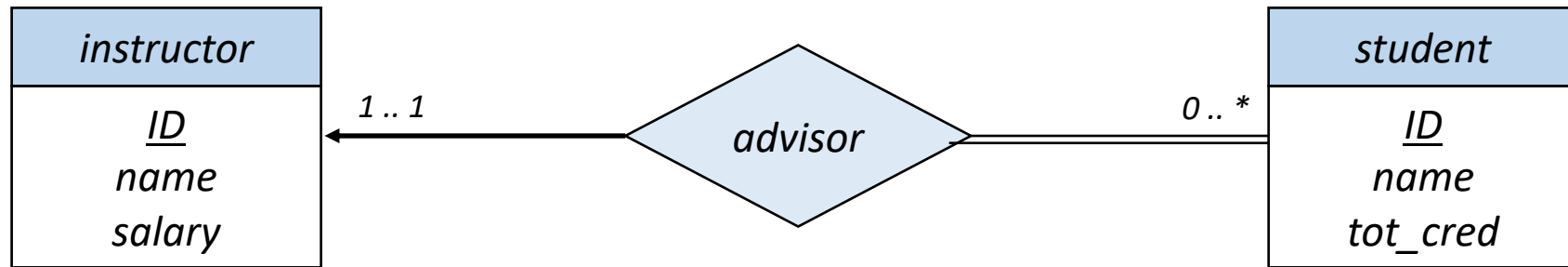


- Nilai minimum 0 menunjukkan partitipasi total dari himpunan entitas dalam himpunan relasi; yaitu, setiap entitas dalam himpunan entitas terjadi setidaknya dalam satu hubungan dalam himpunan relasi tersebut.
- Nilai maksimum 1 menunjukkan bahwa entitas berpartisipasi dalam paling banyak satu hubungan
- Nilai maksimum * menunjukkan nilai tidak terbatas.





- Pada contoh sebelumnya, garis diantara *advisor* dan *student* memiliki kardinalitas 1..1, yang artinya kardinalitas minimum dan maksimum adalah sama-sama 1.
- Yang berarti setiap student harus memiliki satu advisor tepatnya.
- Batas 0.. * pada garis diantara *advisor* dan *instructor* menandakan bahwa seorang instructor dapat memiliki nol atau lebih student.
- Relasi *advisor* adalah one-to-many dari *instructor* ke *student*, dan selanjutnya keikutsertaan *student* dalam *advisor* adalah Total, artinya seorang *student* harus memiliki seorang *instructor*.





Terima Kasih.