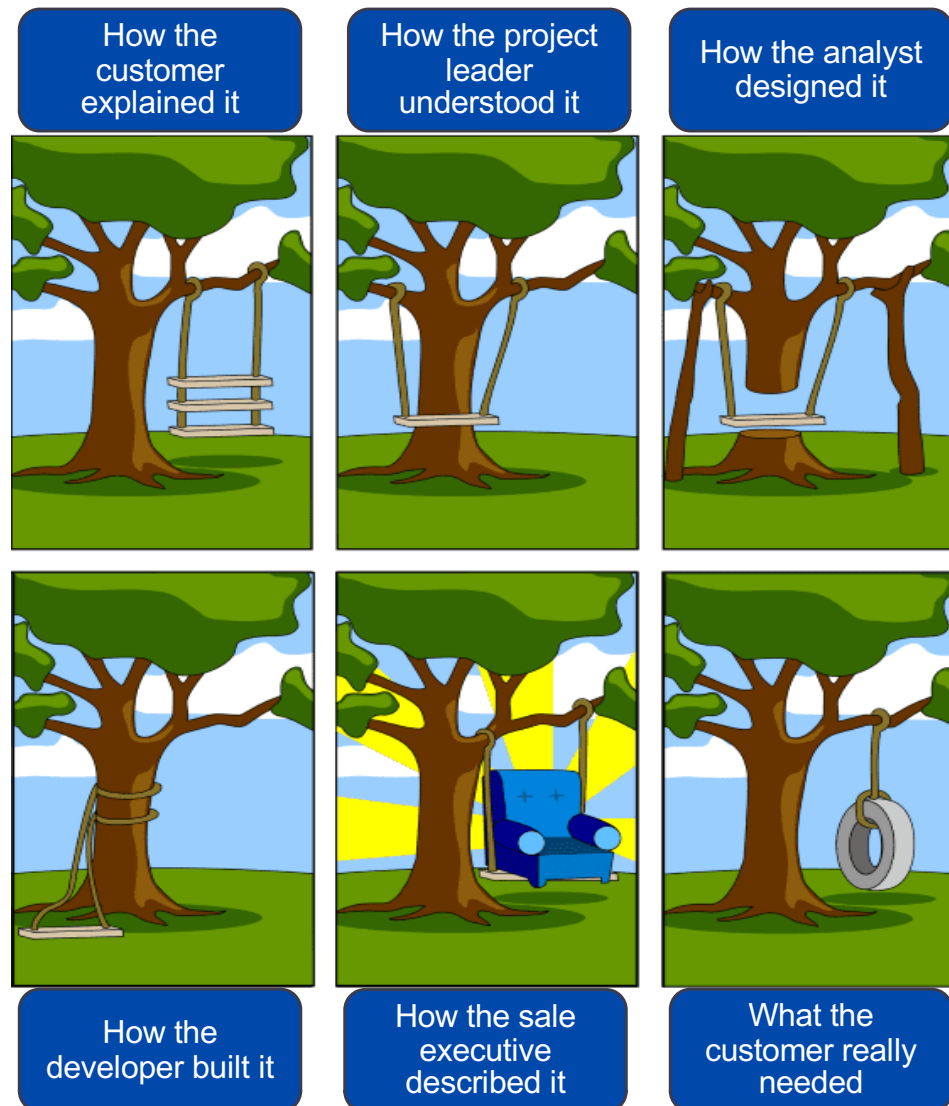# Entity-Relationship (ER) Model

CS 4750
Database Systems

[A. Silberschatz, H. F. Korth, S. Sudarshan, Database System Concepts, Ch.6]
[C.M. Ricardo and S.D. Urban, Database Illuminated, Ch.3]

I've Got Great Ideas for An App

How the customer explained it

How the project leader understood it

How the analyst designed it

How the developer built it

How the sale executive described it

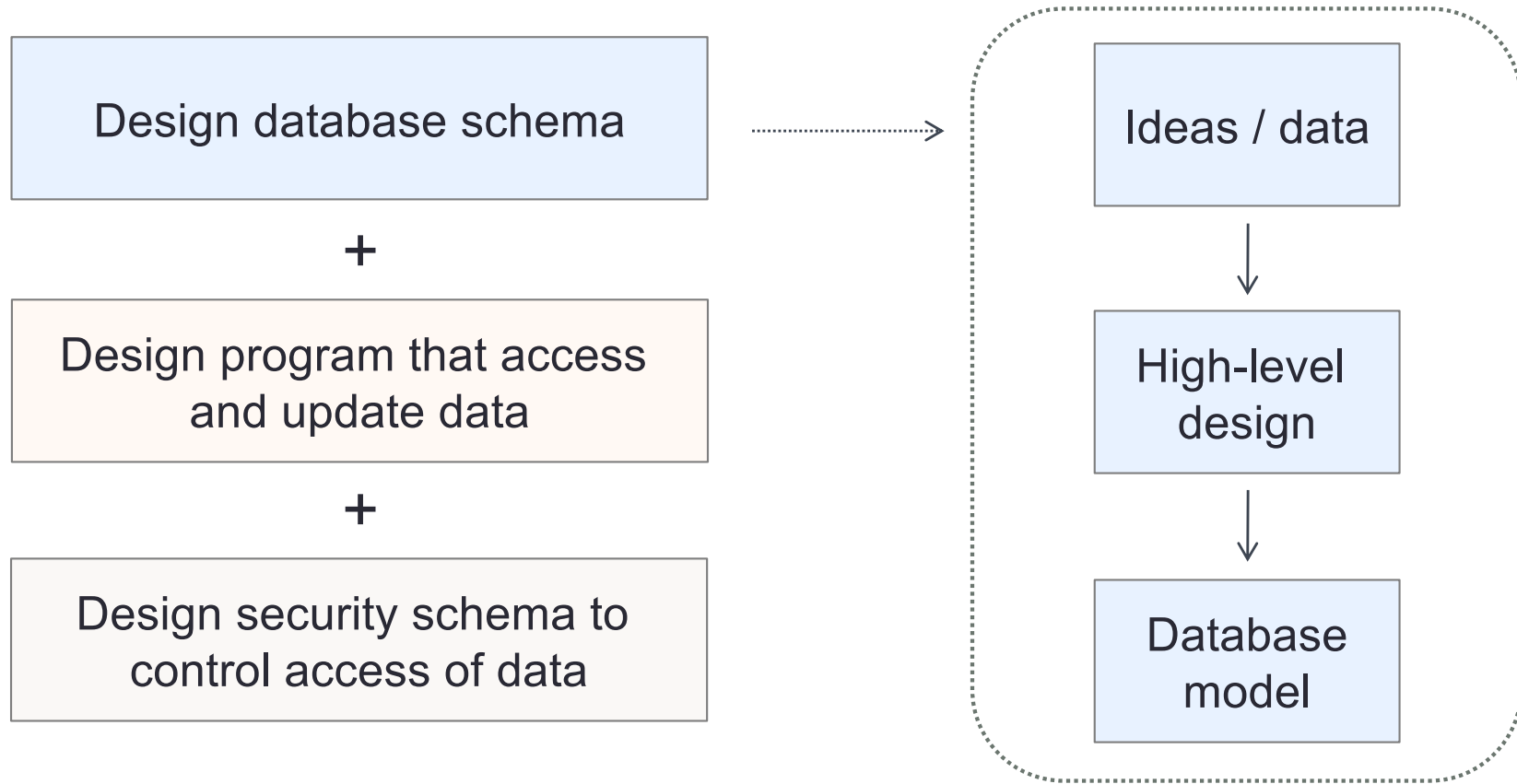**What the customer really needed**

- In database design, communication is key

- Many people are involved in the design process

We need a way to communicate our ideas

[modified from image by Janetti, https://www.cleanpng.com/png-project-management-project-manager-architectural-e-5729993/download-png.html]

Overview: Creating DB App

Design database schema

+

Design program that access and update data

+

Design security schema to control access of data

Ideas / data

↓

High-level design

↓

Database model

Database design = process of organizing data into a database model by considering data to be stored and the interrelationship of the data

Database Design Process

Interact with users and domain experts to characterize the data

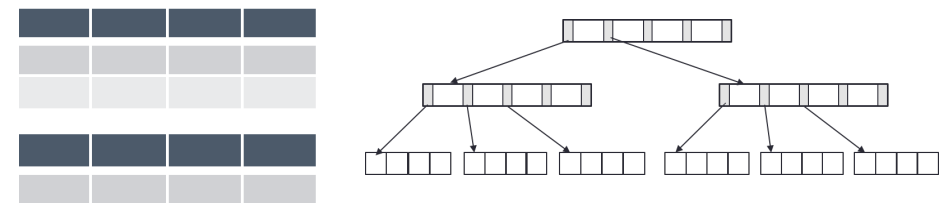Translate requirements into conceptual model (E-R diagrams)
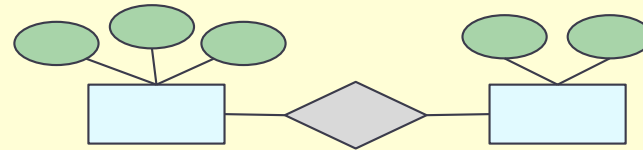
Convert the model to relational model (schema and constraints)

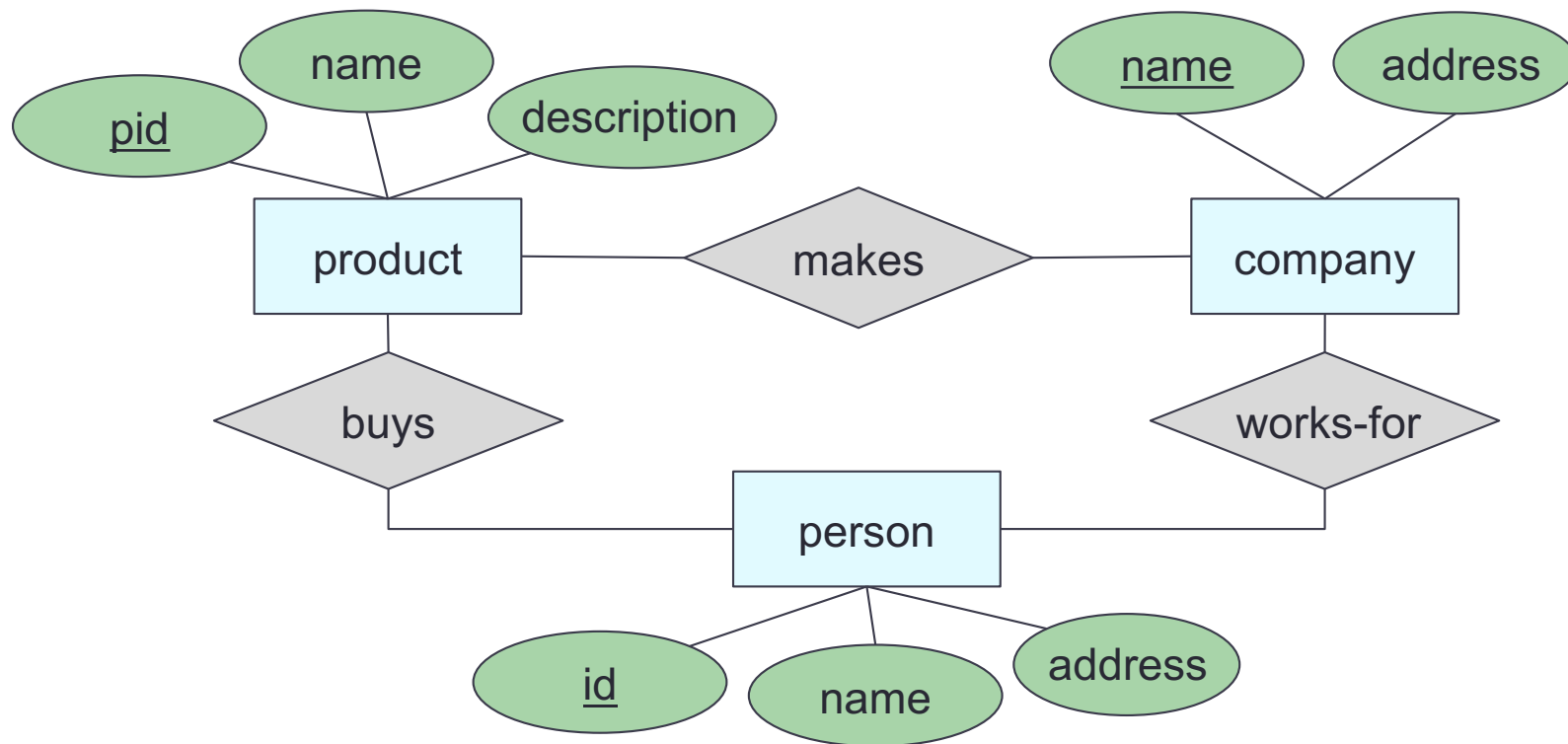Normalize and develop conceptual (logical) schema of the database
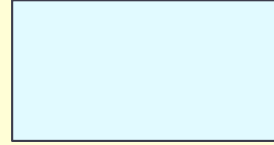
Develop physical schema (partitioning and indexing)

Entity-Relationship Model

- **E-R diagram** – high-level design model representing a database as a collection of entities and relationships among entities
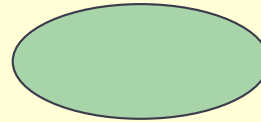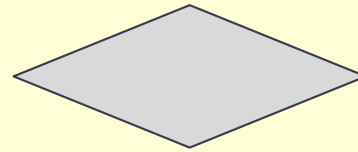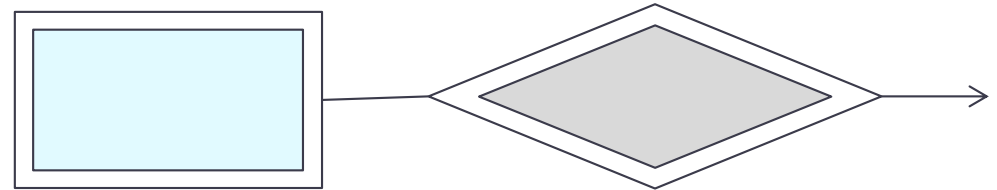
E-R Diagram: Building Blocks
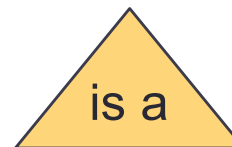
(strong) Entity set
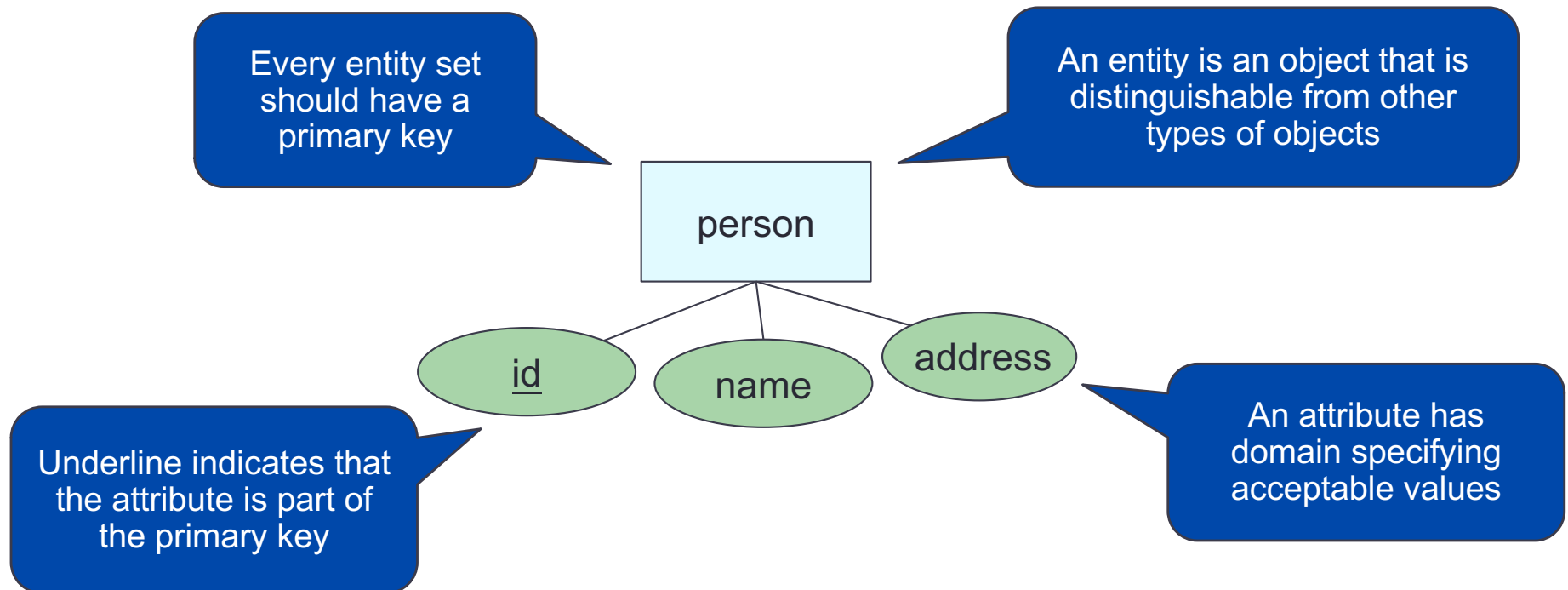
Attribute

Relationship

Weak entity

Subclass

is a

Note: colors are not part of E-R Diagram. They simply are used to increase readability.

Entity Sets and Attributes

- **Entity** ~ an object (~ a row in a table)
- **Entity set** ~ a class (~ table, a collection of similar entities)
- **Attribute** ~ a field (property of the entities in that set)

Every entity set should have a primary key

An entity is an object that is distinguishable from other types of objects

person

id   name   address

Underline indicates that the attribute is part of the primary key

An attribute has domain specifying acceptable values
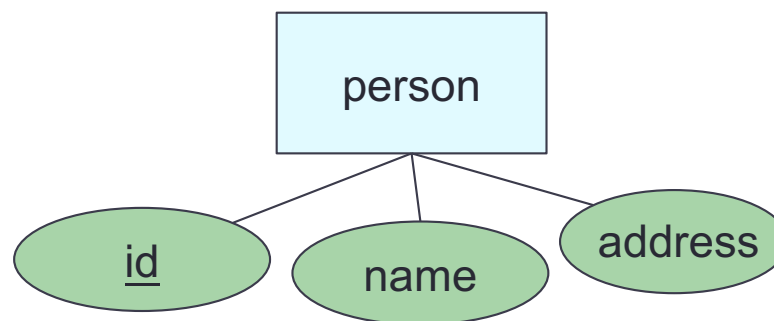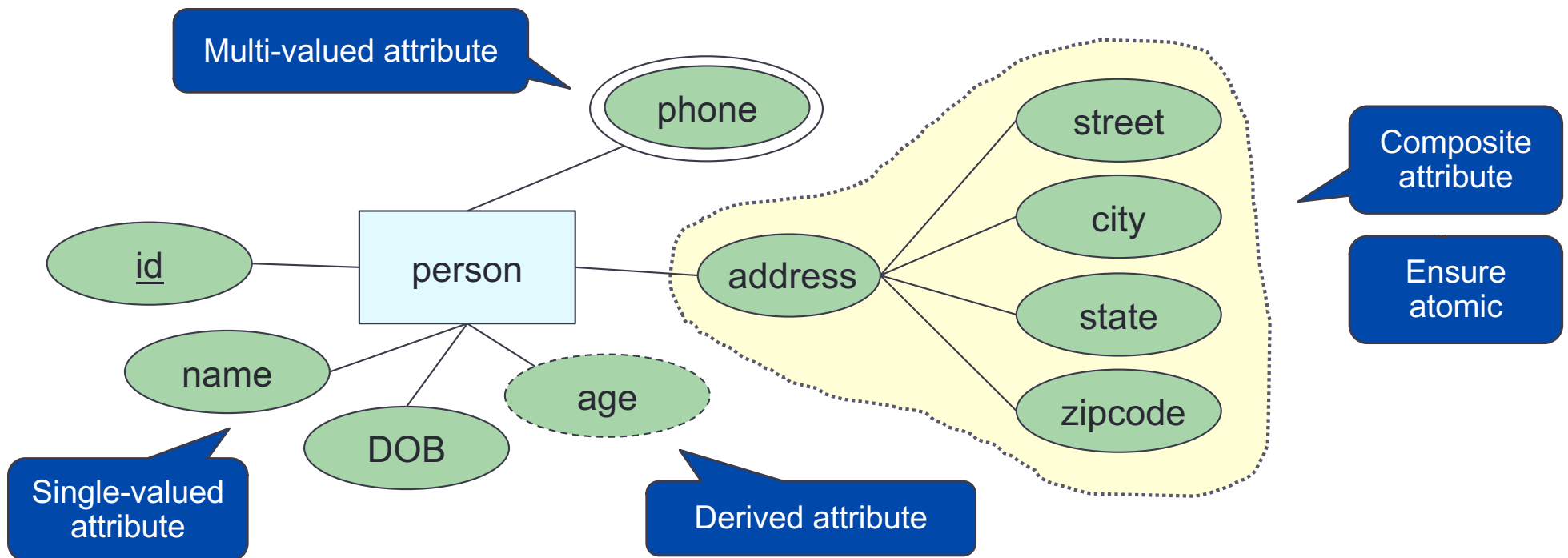
Entity Sets and Attributes

- **Entity** ~ an object (~ a row in a table)
- **Entity set** ~ a class (~ table, a collection of similar entities)
- **Attribute** ~ a field (property of the entities in that set)

> E-R model is a static concept, involving the structure of data and not the operations on data.
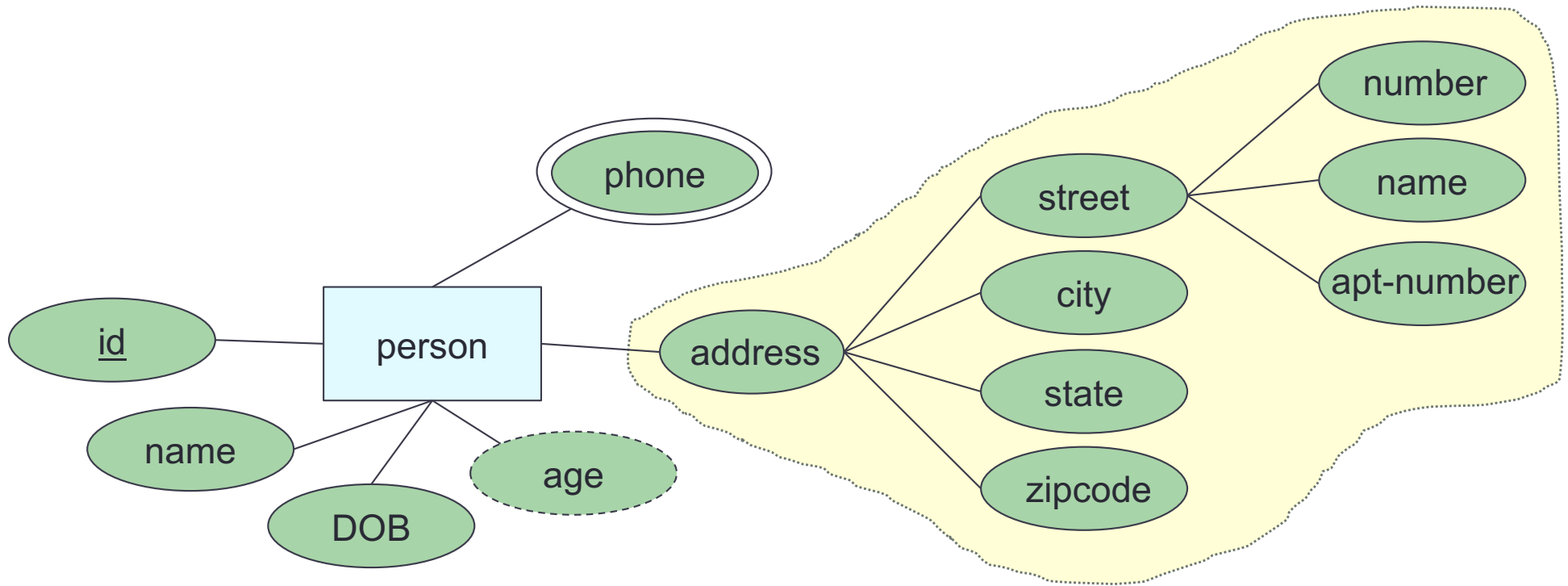> Thus, no methods associated with an entity set

Attributes

- **Single-valued attribute** – allows a single value
- **Multi-valued attribute** – allows multiple values at the same time
- **Derived attribute** –can be calculated from one or more attributes
- **Composite attribute** – consists of multiple values

Attributes

- **Composite attribute** – can have as many levels as needed

Let's try: Movie-Database

**Identify entity, entity sets, and attributes**

Object

Movies

Each movie has a *title* and *year*; *title* and *year* together uniquely identify the movie. *Length* and *genre* are maintained for each movie. Each movie is associated with a *studioName* which tells us the studio that owns the movie, and *producerC#* which is an integer that represents the producer of the movie.

Object

MovieStars

MovieStars tells us something about stars. It maintains the *name* of the movie star, address, gender, and birthdate. The gender can be a single character (*M* or *F*). Birthday is of type "date," which might be a character string of a special form.

Let's try: Movie-Database
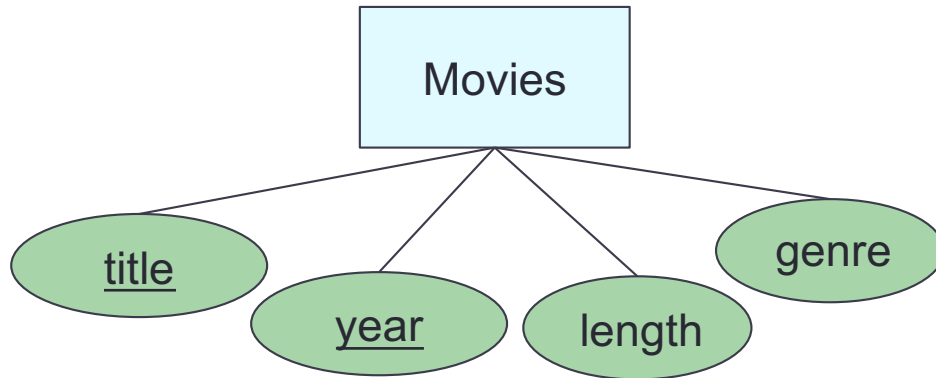
Movies

Title	Year	Length	Genre
Mickey's Club House	1939	231	drama
Awesome Minnie	1977	124	sciFi
Most wanted	1992	95	comedy
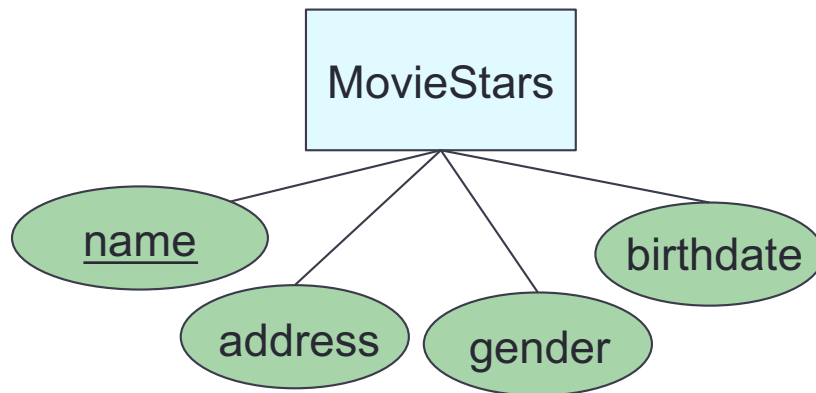
MovieStars

name	address	gender	birthdate
Mickey	11 Somewhere, Charlottesville, VA, 22903	F	01/01/1911
Minnie	22 Another place, Fairfax, VA, 22030	M	02/02/1912
Donald	33 Nowhere, Charlottesville, 22911	F	03/03/1913

Note: instances do not exist in E-R. These tables are only to help visualize the database being designed.

Note: Instances of E-R Diagram

E-R model is used to design a database. The database is not implemented. Therefore, the instance of E-R diagram never exists in the sense that a relation's instances exist in a relational model.

However, it is often useful to visualize the database being designed as if it existed.

Relationships

- Connections among two or more entity sets

- **Binary relationships** – connections between two entity sets

- **Multi-way relationships** (**u-ary**) – connections involving more than two entity sets
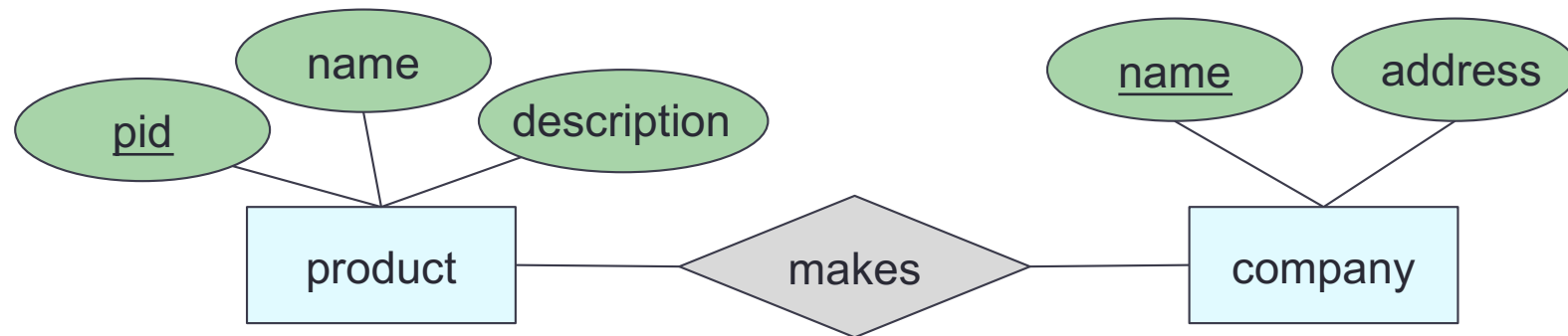
Binary Relationships

If *A* and *B* are sets, a relationship *R* is a subset of *A X B*

product

pid	name	description
11	Beyblade	…
22	Trolls	…
…	…	…

company

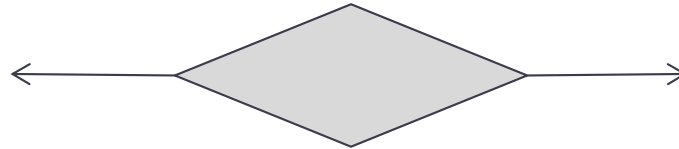| name	address
Dreamwork	…
Hashbro	…
Nyform	…

Note: instances do not exist in E-R. These tables are only to help visualize the database being designed.
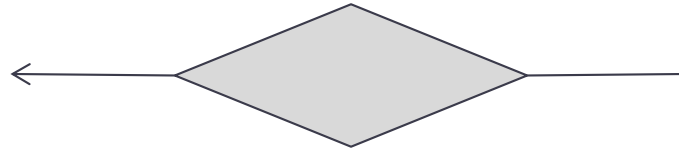
Cardinality (or Multiplicity)

A binary relationship can connect any member of one of its entity sets to any number of members of the other entity set
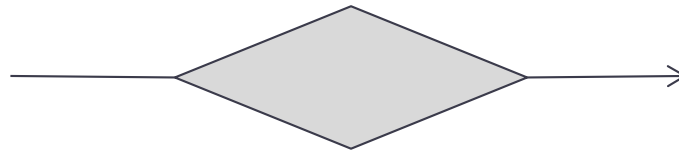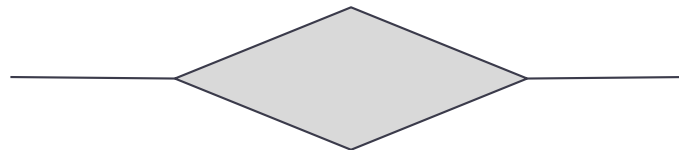
- One-to-one

- One-to-many

- Many-to-one

- Many-to-many

Cardinality: One-to-One

product

pid	name	description
11	Beyblade	…
22	Trolls	…
…	…	…

company

name	address
Dreamwork	…
Hashbro	…
Nyform	…

product ← makes → company

Arrow pointing to the entity that is constrained to one

Arrow pointing to the entity that is constrained to one

Each product can be made by at most one company.
Each company can make at most one product.

"at most one" – Guarantee existence?

Note: instances do not exist in E-R. These tables are only to help visualize the database being designed.
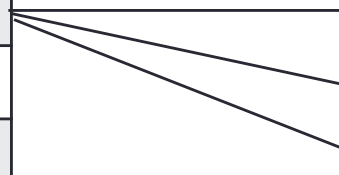
Cardinality: One-to-Many

product

pid	name	description
11	Beyblade	…
22	Trolls	…
…	…	…

company

name	address
Dreamwork	…
Hashbro	…
Nyform	…

product ← makes → company

Arrow pointing to the entity that is constrained to one

Each product can be made by many companies.
Each company can make at most one product.

"at most one" and "many" – Guarantee existence?

Note: instances do not exist in E-R. These tables are only to help visualize the database being designed.

Cardinality: Many-to-One

product

pid	name	description
11	Beyblade	…
22	Trolls	…
…	…	…

company

name	address
Dreamwork	…
Hashbro	…
Nyform	…

product — makes → company

Arrow pointing to the entity that is constrained to one

Each product can be made by at most one company.
Each company can make many products.

"at most one" and "many" – Guarantee existence?

Note: instances do not exist in E-R. These tables are only to help visualize the database being designed.
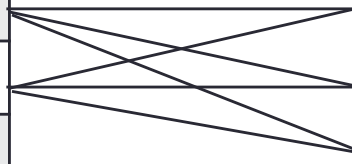
Cardinality: Many-to-Many

product

pid	name	description
11	Beyblade	…
22	Trolls	…
…	…	…

company

| name	address
Dreamwork	…
Hashbro	…
Nyform	…

product — makes — company

Each product can be made by many companies.
Each company can make many products.

"many" – Guarantee existence?

Note: instances do not exist in E-R. These tables are only to help visualize the database being designed.
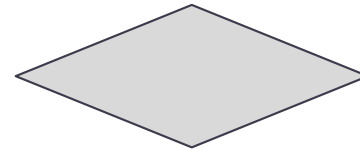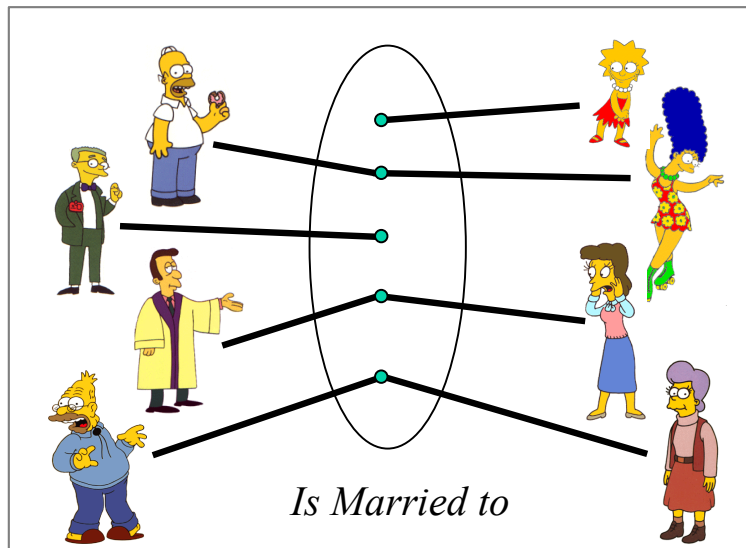
Let's try: Cardinality (1)

Draw an E-R diagram to model the given image sample data. Specify the cardinality. Interpret the E-R diagram.
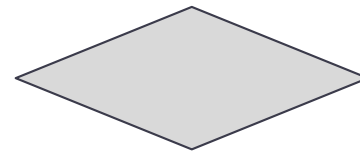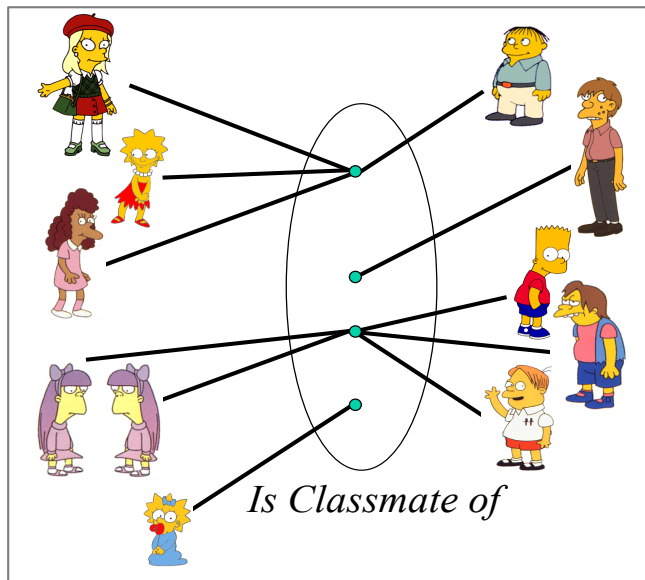

Was born in

was_born_in
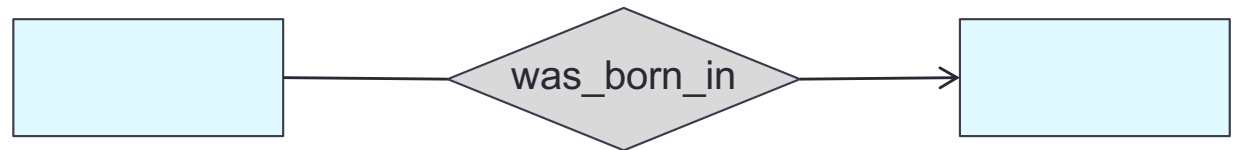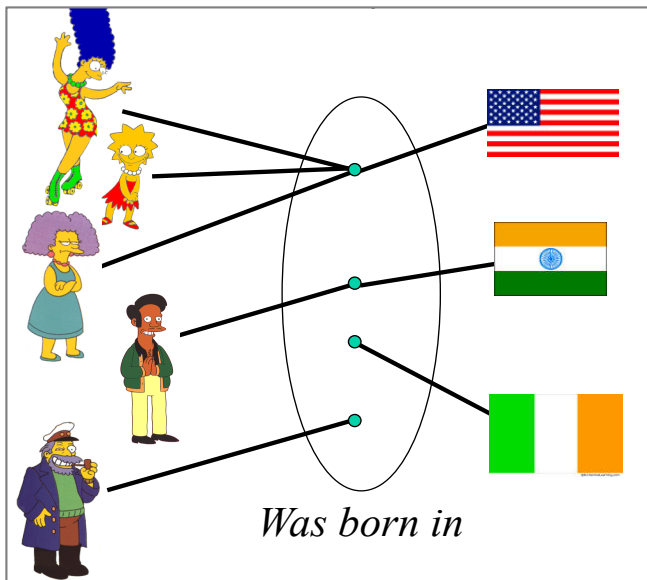
(many-to-one)

Let's try: Cardinality (2)

Draw an E-R diagram to model the given image sample data. Specify the cardinality. Interpret the E-R diagram.
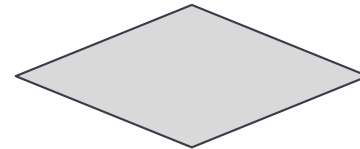

Is Married to

Let's try: Cardinality (3)

Draw an E-R diagram to model the given image sample data. Specify the cardinality. Interpret the E-R diagram.

Is Classmate of

Let's try: Cardinality (4)

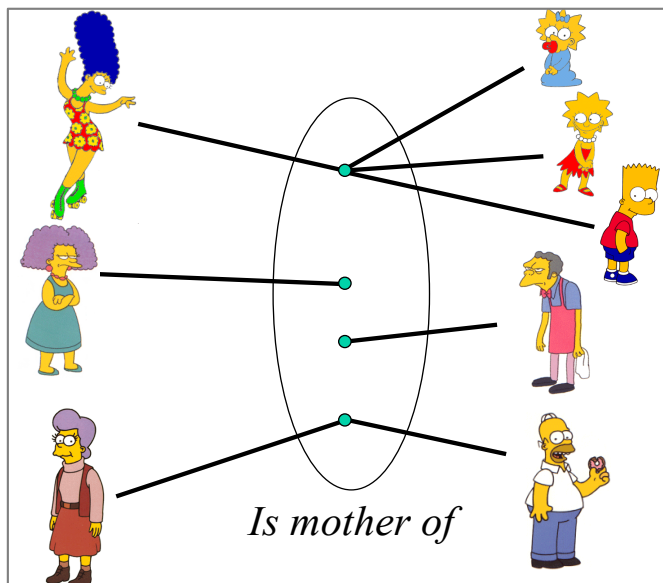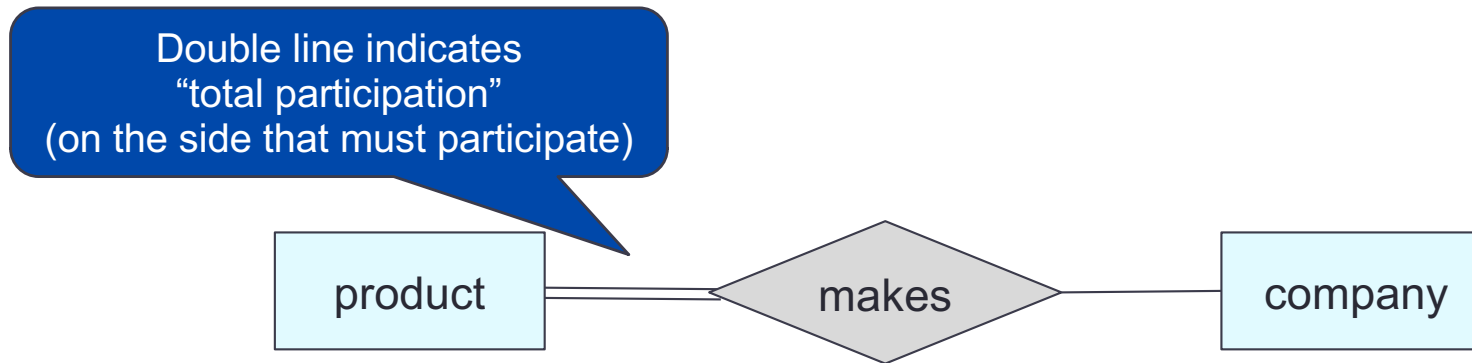**Draw an E-R diagram to model the given image sample data. Specify the cardinality. Interpret the E-R diagram.**

Total Participation

Total participation – all entities in an entity set must participate in the relationship

Double line indicates "total participation" (on the side that must participate)

product — makes — company

Every product must be made by at least one company.

Each product can be made by many companies.

Each company can make many products.

Some companies may not make any product.

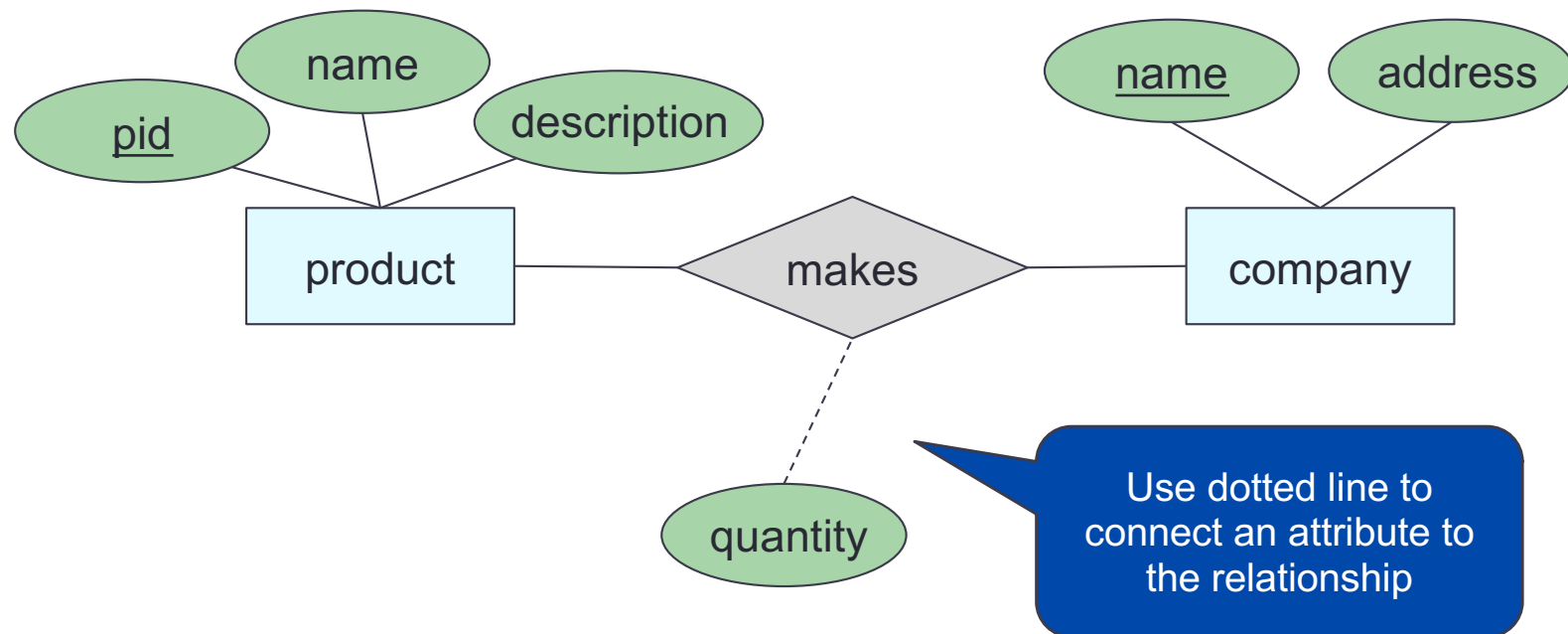# Let's try: Participation (1)

Interpret the E-R diagram.

Interpret the E-R diagram.
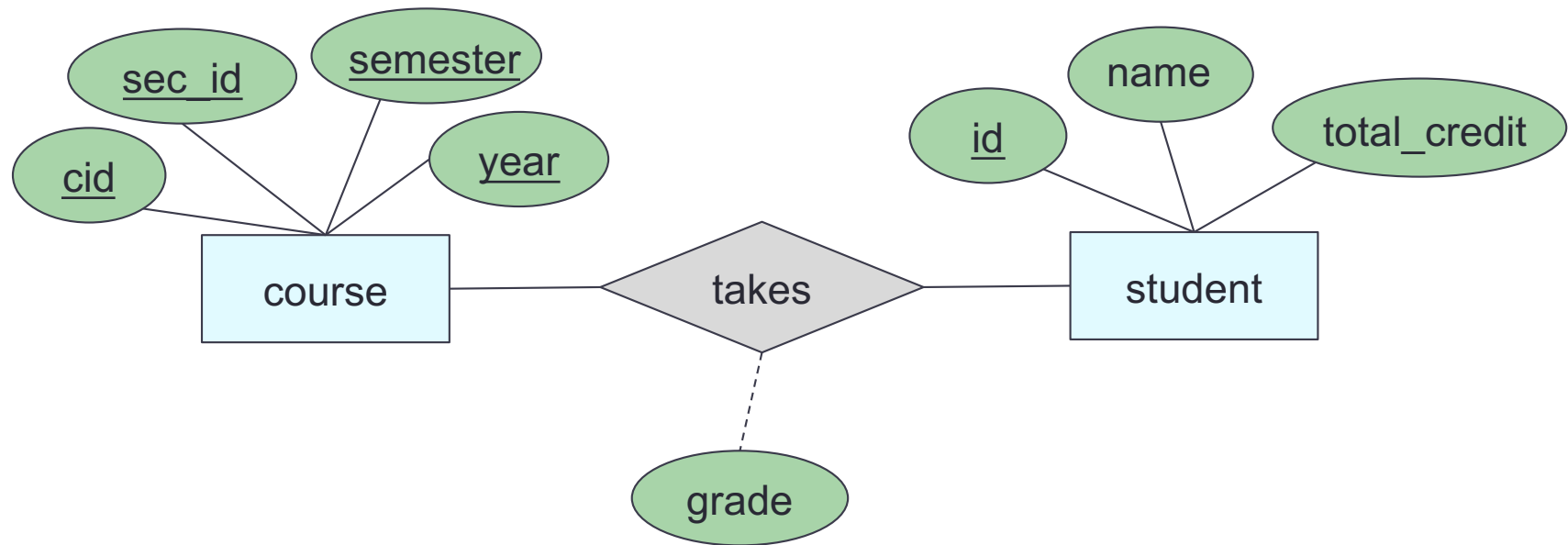

employee —— assigned to —— department

Attributes on Relationships

- Relationships can have attributes

- The attributes have values only when the relationship occurs

Let's try: Attributes - Relationships

Interpret the E-R diagram.

Wrap-Up

• Database design process

• Intro to E-R model

• Entities and entity sets

• Attributes: single-valued, multi-valued, derived, composite

• Cardinality and participation

What's next?

• Roles in relationships

• Relationships: binary, u-ary

• Weak entity

• Subclassing

• E-R to relational model

Practice Activity: E-R Diagram

Pair any of the following entity sets. Come up with as many relationships as you can between the pairs.

Specify the cardinality. Identify any mandatory relationships (total participation)

Person	Car
Student	Office
Manager	Pizza
Employee	Laptop
Cat	Homework
Dog	Exam
Textbook	Hotdog
Course	Fruit