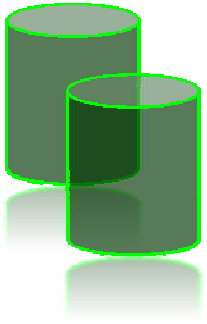


Database Systems
Part-1

Introduction and ER Model

Arijit Khan

Associate Professor
Department of Computer Science
Aalborg University, Denmark

arijitk@cs.aau.dk

Lecturers

- Arijit Khan
- arijitk@cs.aau.dk
- Teaches till the week of Oct 10 (Week 41)

- Matteo Lissandrini
- matteo@cs.aau.dk
- Teaches after the the week of Oct 10 (Week 41)

Teaching Assistants

- Jan Mackeprang Damgaard Hansen (jmdh19@student.aau.dk)
- Martin Pekár Christensen (mpch@cs.aau.dk)
- Abiram Mohanaraj (abiramm@cs.aau.dk)
- Ghadeer Abuoda (gsmas@cs.aau.dk)

Introduction

- Course website

Moodle:
https://www.moodle.aau.dk/course/view.php?id=44380

- All course materials on Moodle

(slides, exercise, self-study/ mini-project, quiz, . . .)

Textbooks

- Database Systems: The Complete Book

Hector Garcia-Molina, Jeffrey D. Ullman, Jennifer Widom

- Database System Concepts 7th edition

Avi Silberschatz, Henry F. Korth, S. Sudarshan

https://factumbooks.dk/?booklist=3&id=3307

My Teaching Style

- Slides override book
- There will be many **classroom exercises** throughout all lectures
- Questions will be welcomed
- There will be some **study-home slides** throughout most lectures

Quizzes

- One quiz for each lecture on Moodle
- Recommended to do a couple of days after the lecture
- Unlimited number of attempts

Exercise

- Deepen understanding and application of the material covered in the previous week
- Discussions plus training for the written exam
- Mind the notation(!) – avoid losing points in the exam
- TAs will visit you in the group rooms during the scheduled exercise sessions

Questions?

Self-study (mini-project)

- Mini-project divided into 2 parts
  - Database modeling and normalization
  - Loading and querying

- Work in groups of 8-10 students. Register your group to TAs by Sep 23's exercise session.

- Part-1 of self-study to release on Sep 26.

- Complete Part-1 of self-study by Oct 3. Send to peer-group and TA by Oct 3's exercise session.

- Receive feedback by peer-group and TA on Oct 10 (Lecture session).

Questions?

Exam

- Written exam (physical)
- 1 hour
- No books, notes, electronic devices, etc.
- Just brain, pen, and paper
- 100 points
- Passing limit: 50 points

➢ Format might change depending on pandemic, other factors

Schedule for first half

Intro & ER	Sep 5 Monday (12:30-14:15), Lecture - FRB 7H	No exercise
Relation, FD, Keys	Sep 12 Monday (12:30-14:15), Lecture - FRB 7H	Exercise-1 on Intro & ER (14:30-16:30)
Normalization	Sep 19 Monday (12:30-14:15), Lecture - FRB 7H	Exercise-2 on Relation, FD, Keys (14:30-16:30)
Relational Algebra	Exercise-3 on Normalization (8:15-10:00)	Sep 23 Friday (10:15-12:00), Lecture - NOVI8
Release of Self-Study-1	Sep 26 Monday, No lecture, No exercise	
SQL – Part I	Oct 3 Monday (12:30-14:15), Lecture - FRB 7H	Exercise-4 on Relational Algebra (14:30-16:30)
Feedback of Self-Study-1	Oct 10 Monday (12:30-14:15), During lecture - FRB 7H	Exercise-5 on SQL – Part I (14:30-16:30)

Ask Questions!

> The important thing is not to stop questioning.
>
> Albert Einstein

But why should we care?

- We are in the era of “big data”
  - Enormous volumes of data emerge from almost every industry
- We need efficient systems to manage such data

Database and DBMS

- What is a database?
  - A collection of related data, specially organized for efficient retrieval by a computer
- What is a database management system (DBMS)?
  - A piece of software that helps us efficiently manage/retrieve information from databases
- Database System (DBS):

  - A database and a DBMS

DBMS in Practice

- Large web sites rely heavily on DBMS
  - Facebook
  - Twitter
- Many non-web companies, too
  - E.g., Banks
- Even small pieces of software on your computer
  - E.g., Google Chrome

Relational Model

- Numerous DBMS exist on the market
  - Oracle, SQL Server, DB2, MySQL…
- Most of them follow the relational model
- What does it mean?
- Answer: They store all data in relations

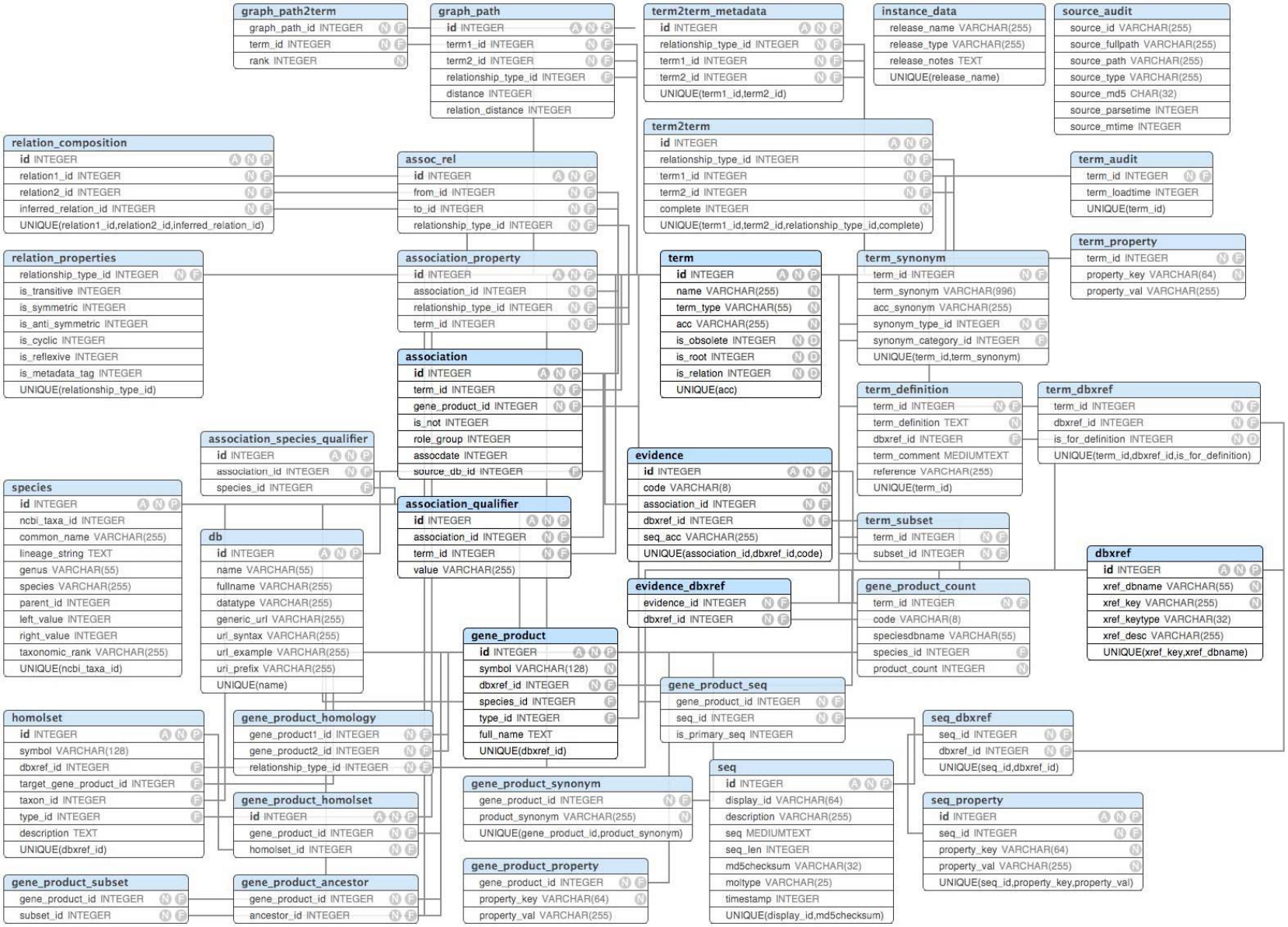
Relation

Product

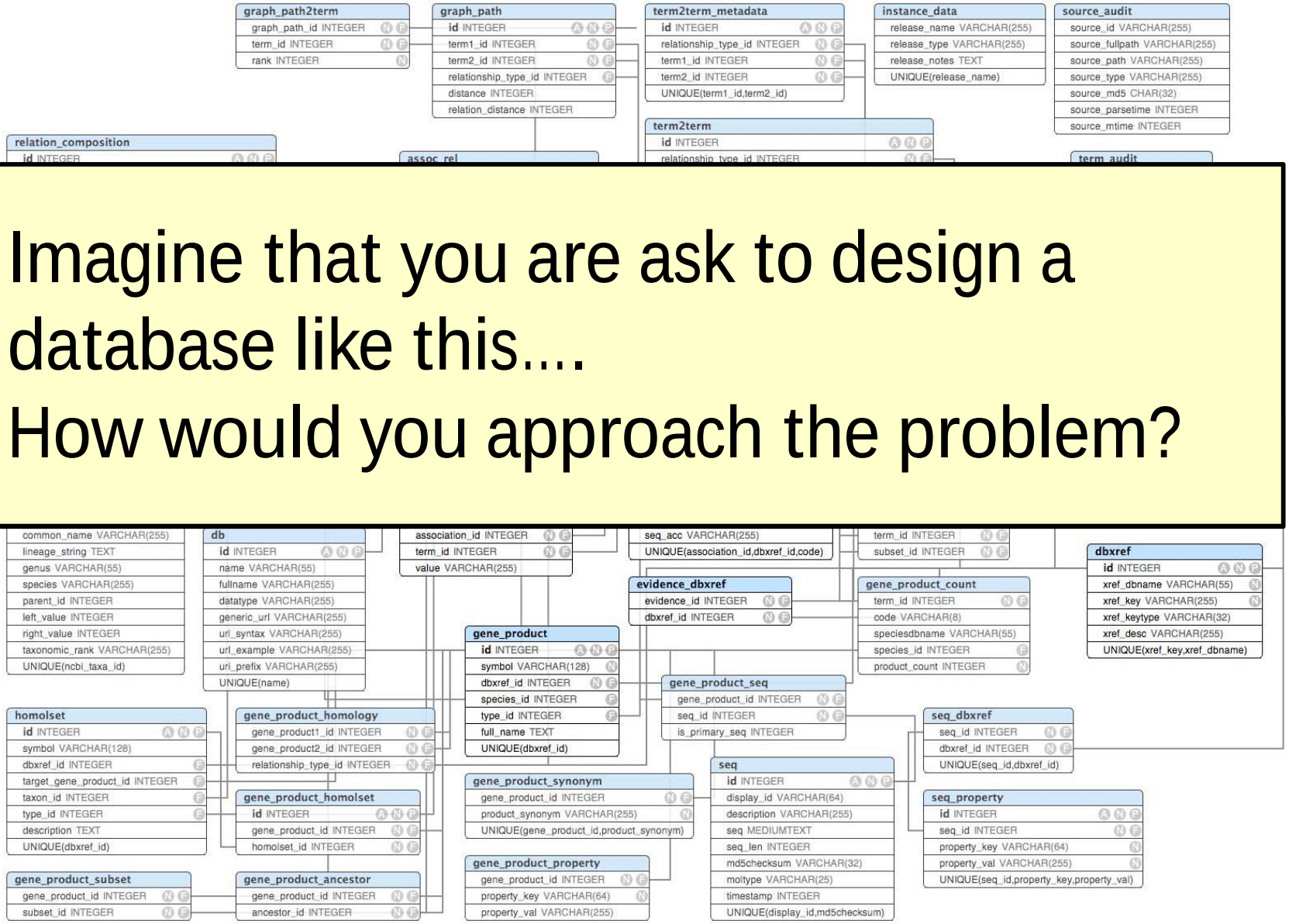
Name	Price	Category	Manufacturer
iPhone 6	888	Phone	Apple
iPad Air 2	668	Tablet	Apple
Galaxy	798	Phone	Samsung
EOS-1D X	1199	Camera	Canon

- Some jargons:
  - A relation is often referred to as a table
  - A row in a table is also called a tuple or a record
  - A column in a table is also called an attribute of the table

A real database may have a large number of tables...



A real database may have a large number of tables...



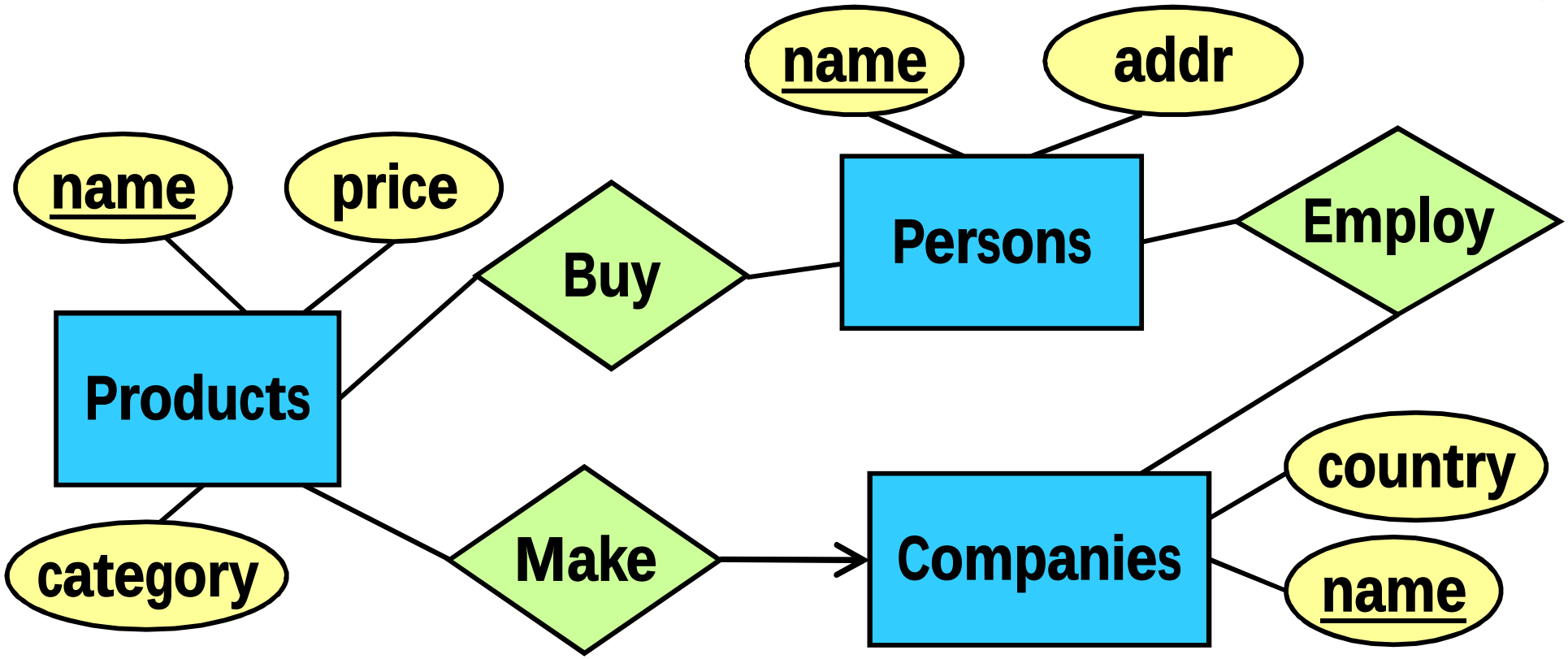
- Imagine that you are ask to design a database like this....
- How would you approach the problem?

Designing a Database for an Application

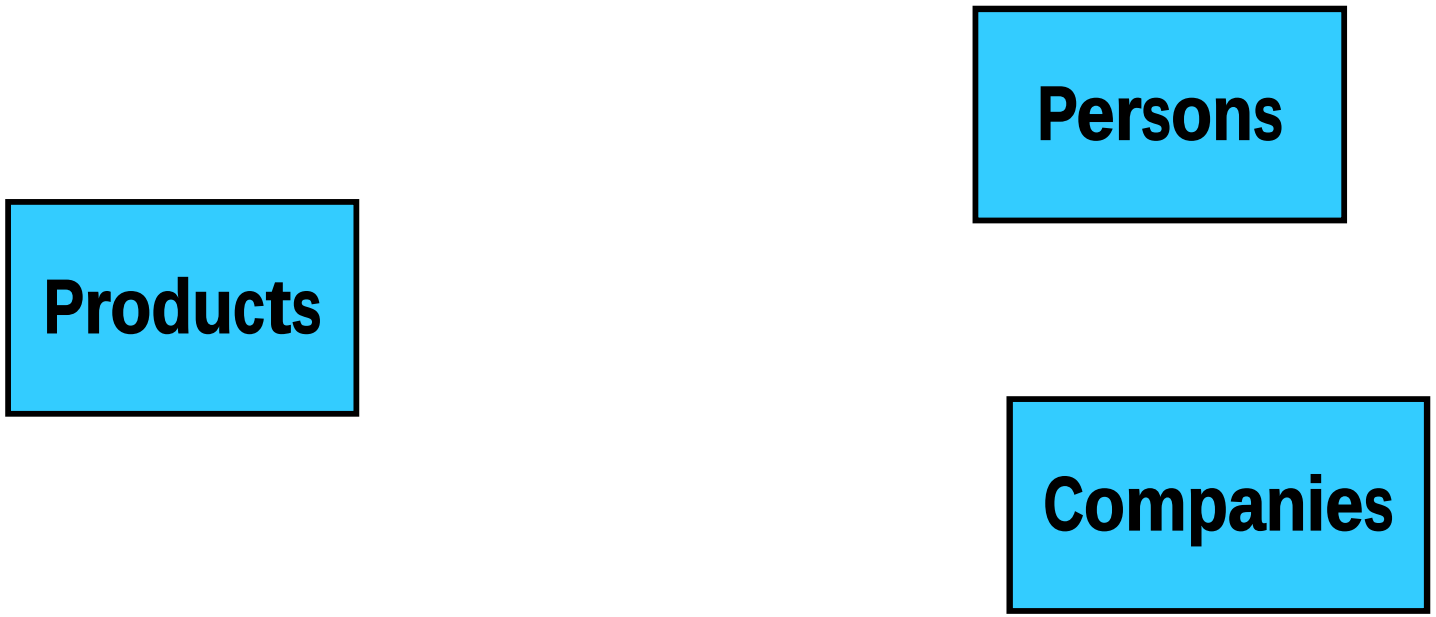
- Conceptually model the requirements of the application
  - What are the things that need to be stored?
  - How do they interact with each other?
- Tool to use: Entity-Relationship (ER) Diagrams
  - A pictorial and intuitive way for modelling
- Translate the conceptual model into a set of tables
- Construct the tables with a DBMS

Questions?

ER Diagram

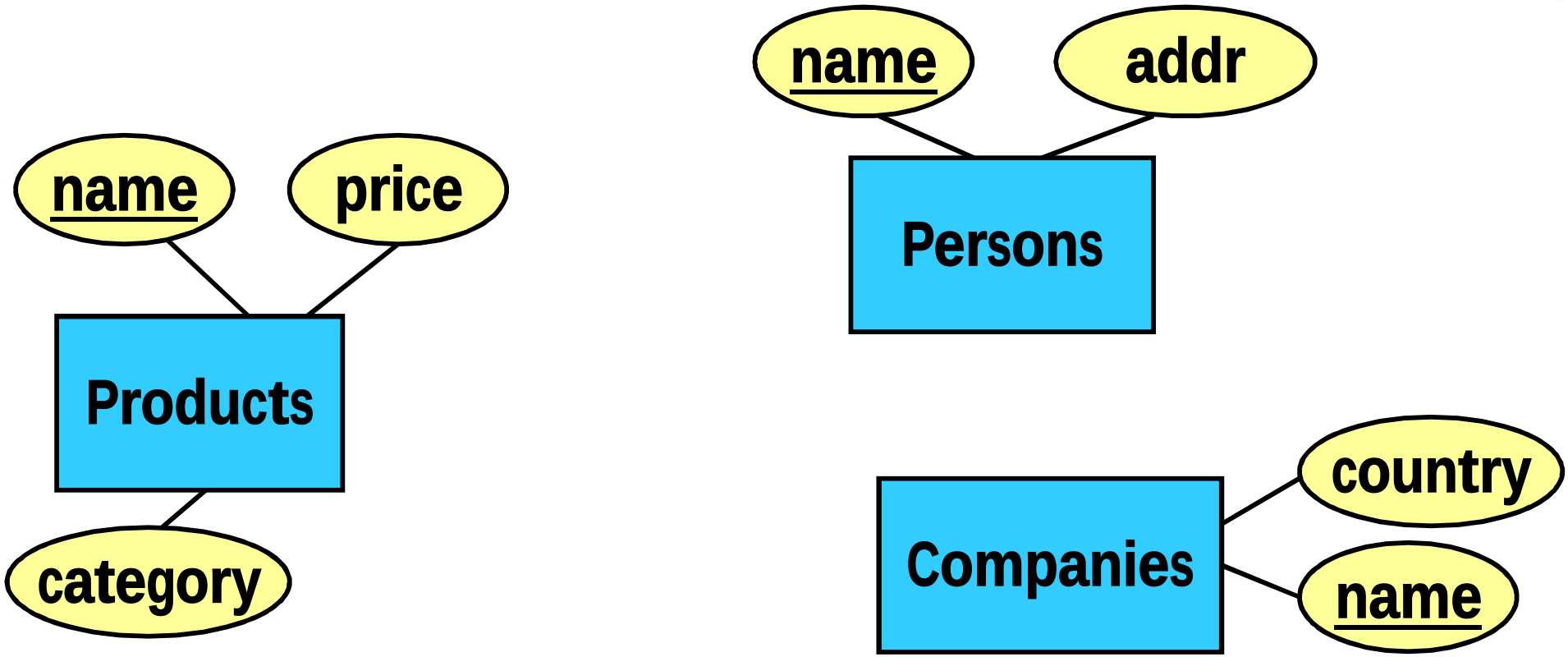

name
addr
name
price
Buy
Persons
Employ
Products
country
Make
Companies
category
name

ER Diagram



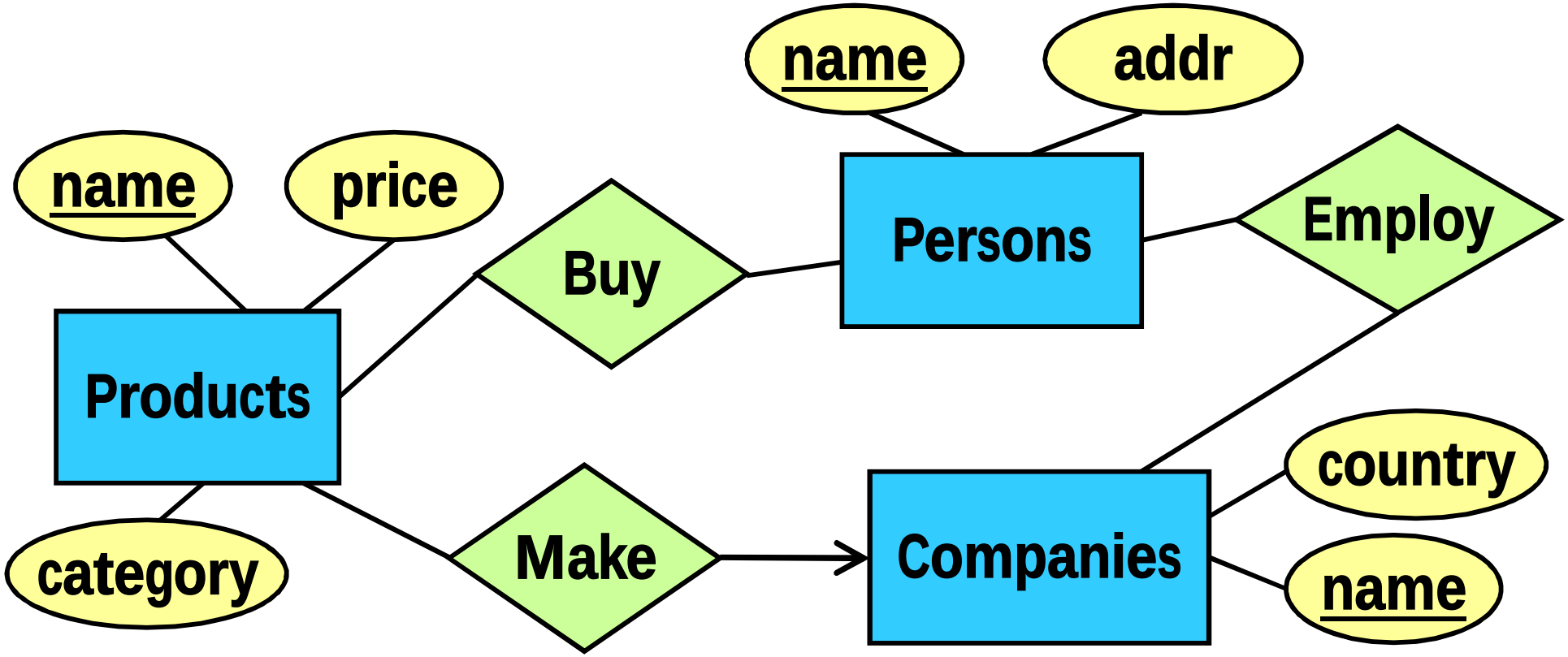
- Rectangle = Entity Set
- Entity = Real-world object (e.g., a bar)
- Entity Set = Collection of similar objects (e.g., a set of bars)
- Analogue: An object class

ER Diagram



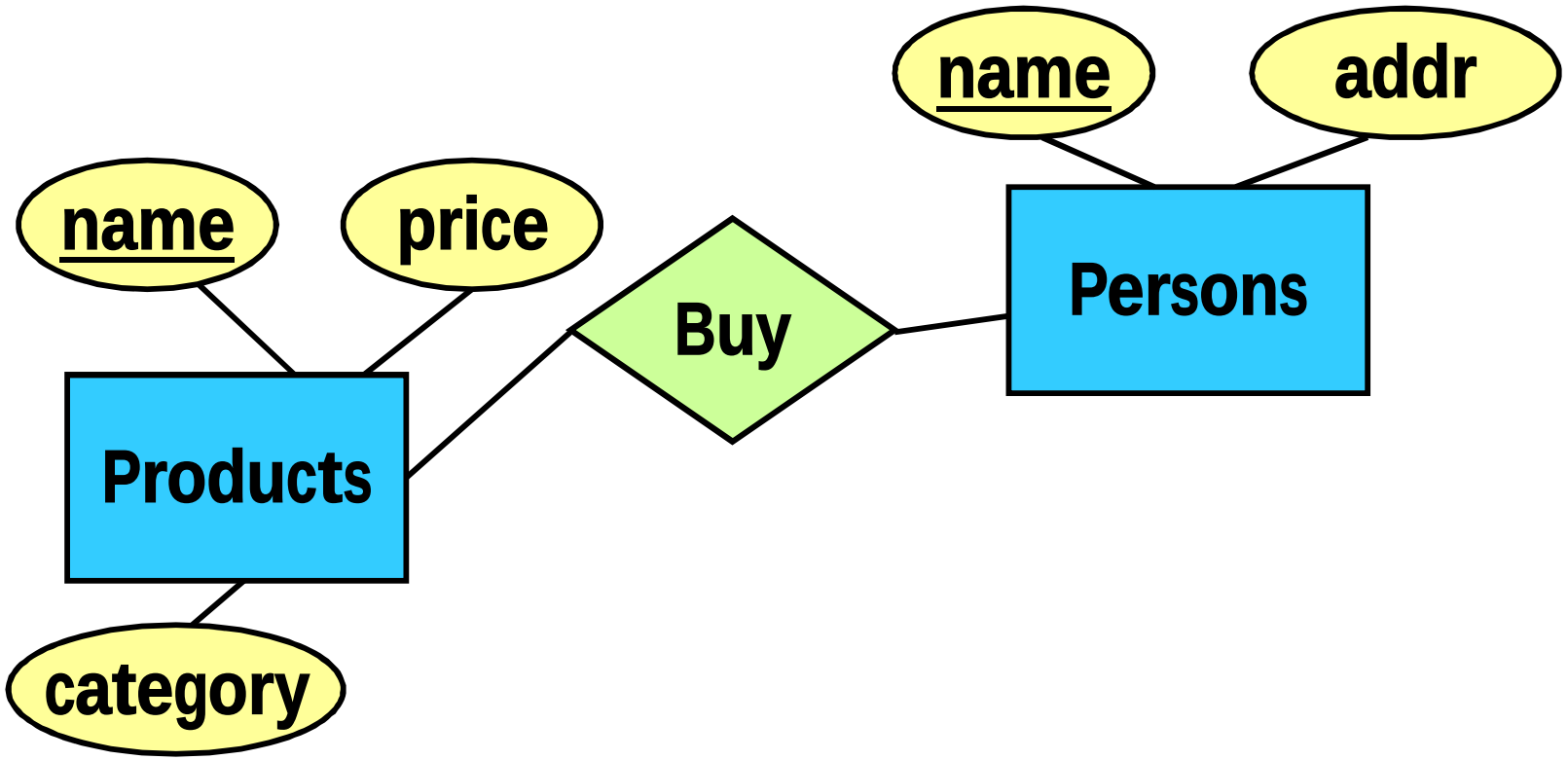
- Oval = Attribute = Property of an entity set

ER Diagram



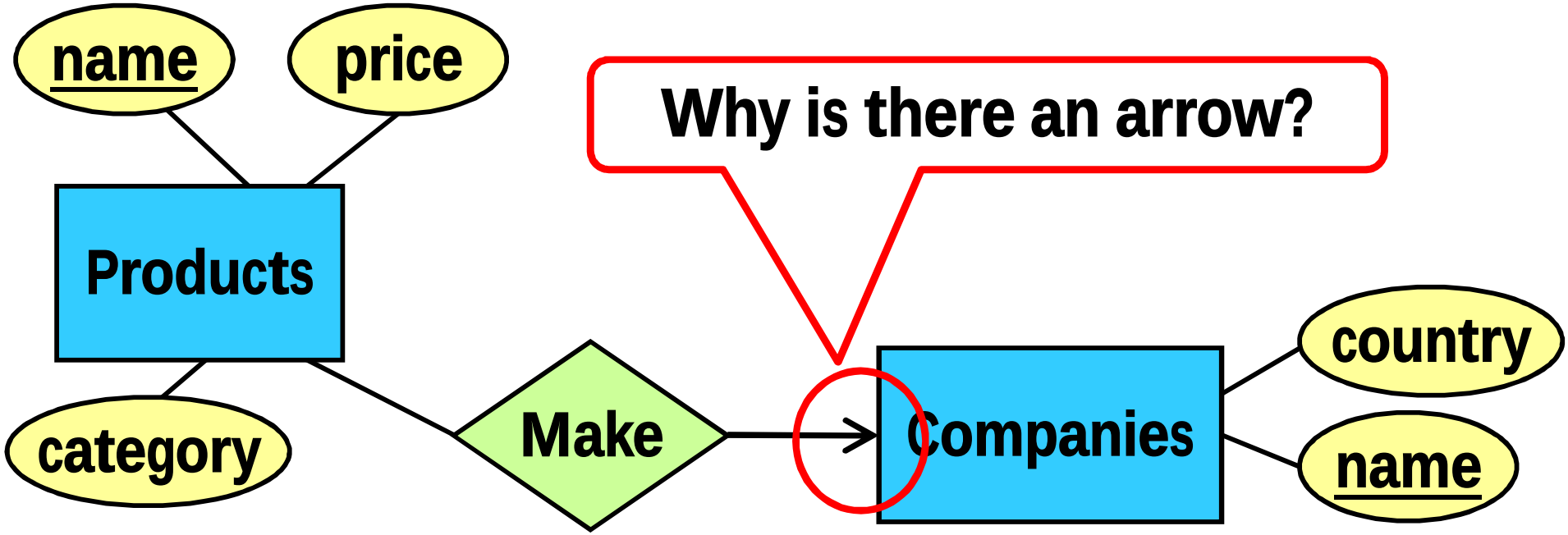
- Diamond = Relationship = Connection between two entity sets

ER Diagram



- Diamond = Relationship = Connection between two entity sets
- Persons buy products

ER Diagram

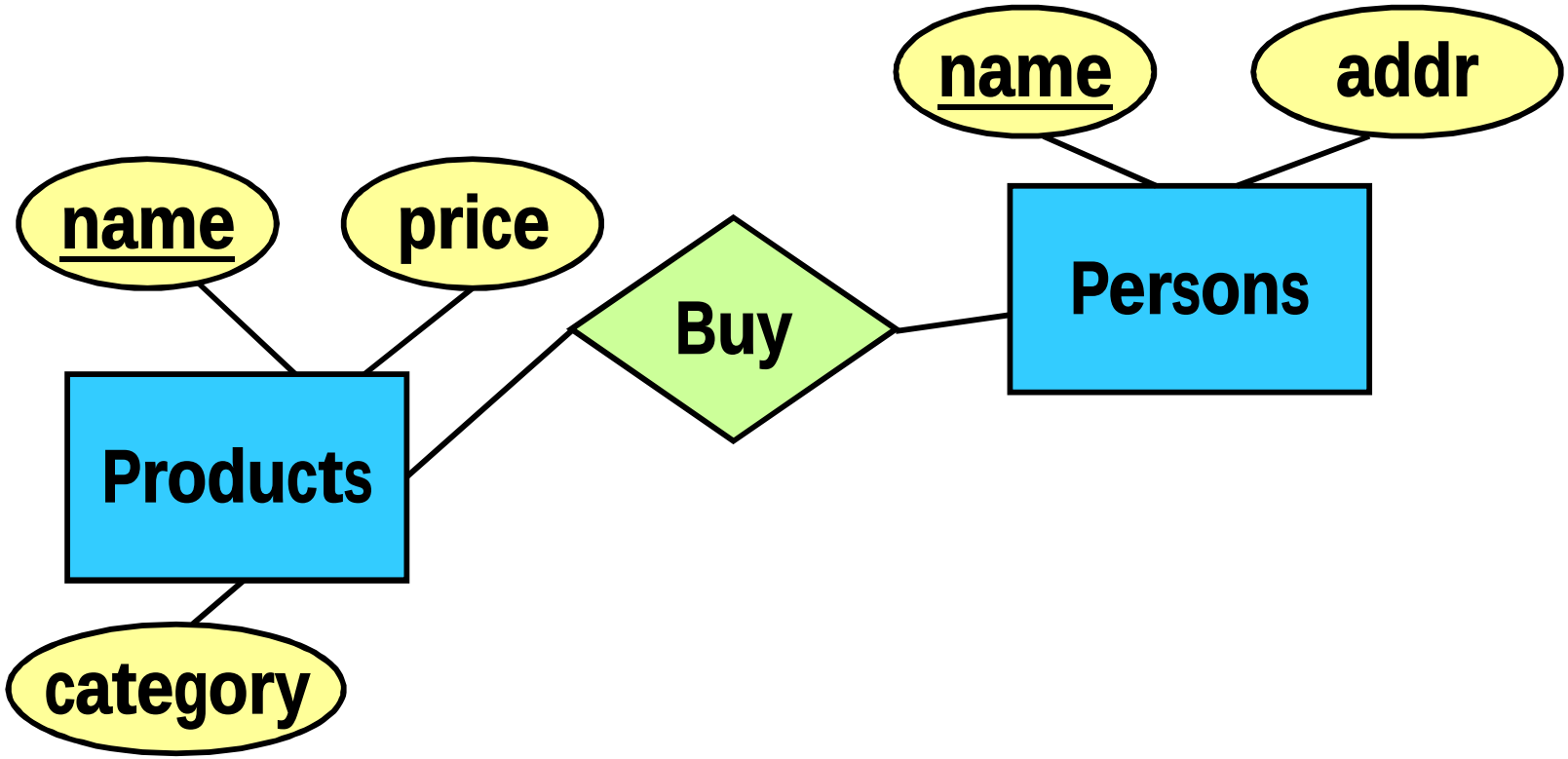


- Diamond = Relationship = Connection between two entity sets
- Companies make products

ER Diagram

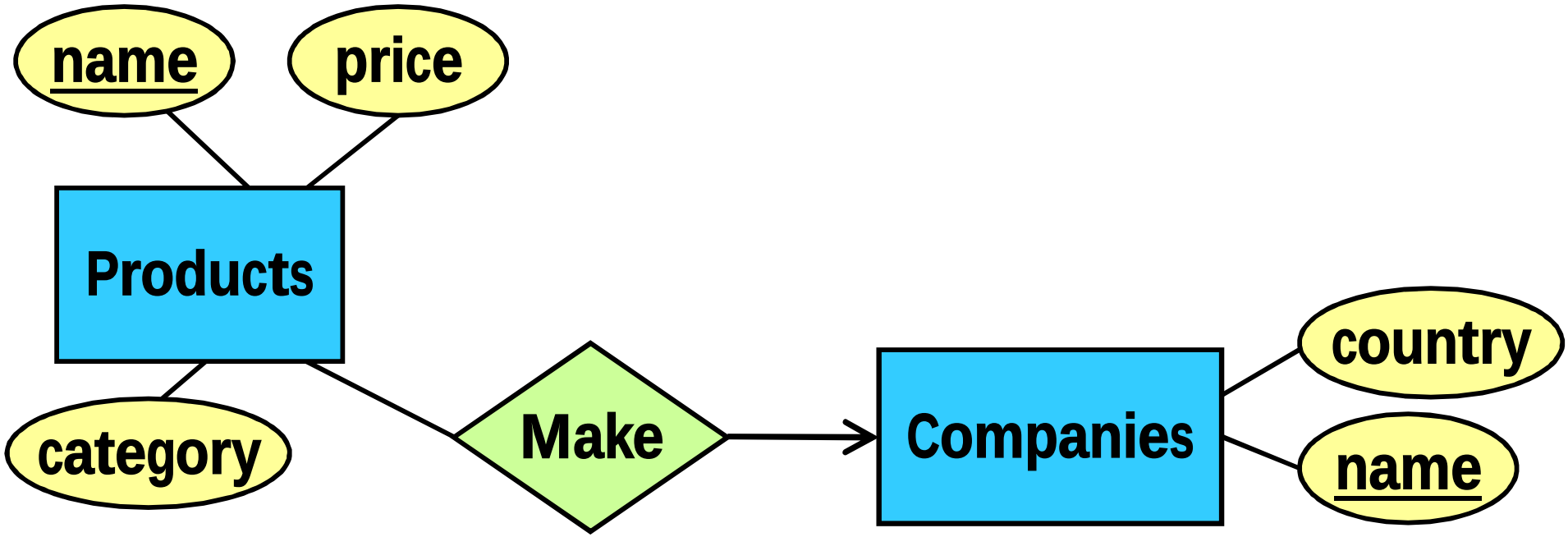
- Many-to-Many Relationships
- Many-to-One Relationships
- One-to-One Relationships

Many-to-Many Relationship



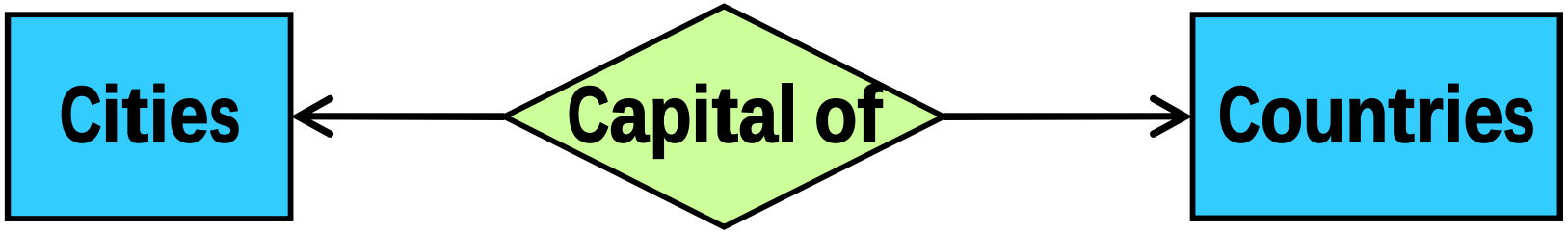
- One person can buy multiple products
- One product can be bought by multiple persons

Many-to-One Relationship



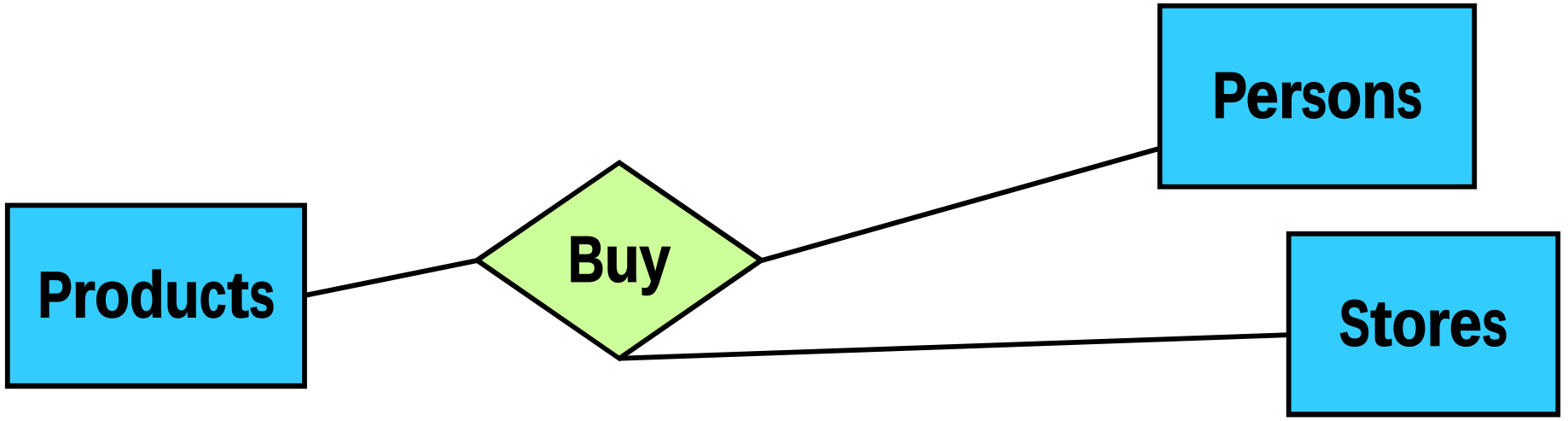
- One company can make multiple products
- But one product can only be made by one company

One-to-One Relationship



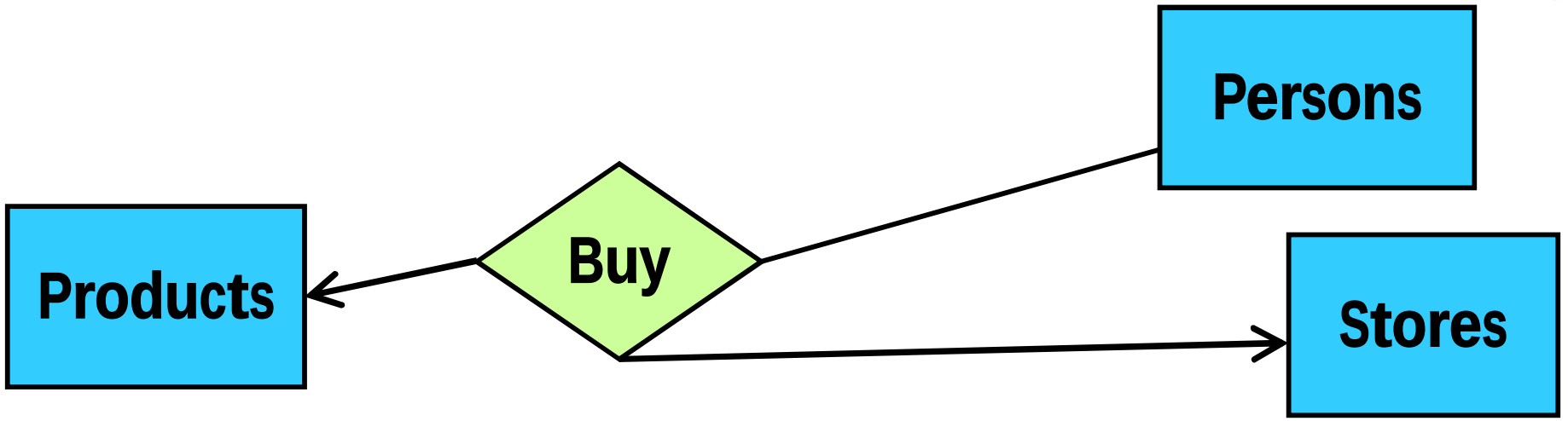
- A city can be the capital of only one country
- A country can have only one capital city

Multi-way Relationships



- What if we want to record the store from which the person buys the product?
- We can use a 3-way relationship
- Drawback: The arrows would be complicated

Multi-way Relationships



- What does this mean?
- From <person, store> to product: many to one
- From <person, product> to store: many to one
- From <product, store> to person: one to many
- Note: arrows with multi-way relation not required in exercise/exam

Questions?

Classroom Exercise-1

- Each player prefers only one game, but not vice versa
- Many-to-many?
- Many-to-one?
- One-to-one?

Classroom Exercise-1: Solution

- Each player prefers only one game, but not vice versa
- Many-to-many?
- Many-to-one?
- One-to-one?

Classroom Exercise-2

- No two shops sell the same product
- Many-to-many?
- Many-to-one?
- One-to-one?

Classroom Exercise-2: Solution

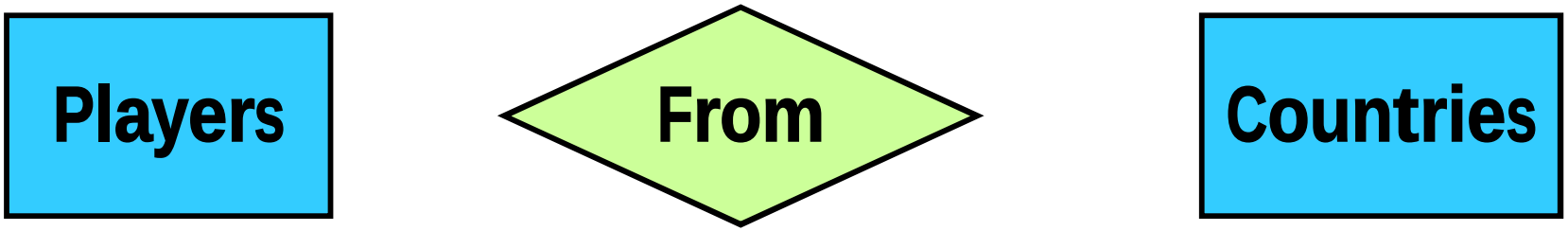
- No two shops sell the same product
- Many-to-many?
- Many-to-one?
- One-to-one?

5 Minutes Interval

The important thing is not to stop questioning.

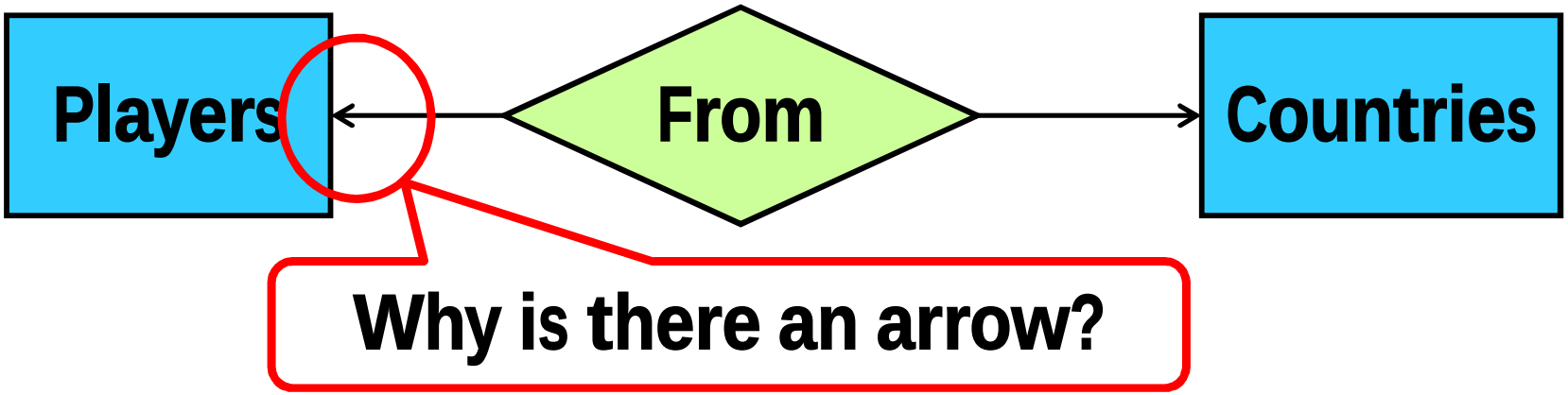
Albert Einstein

Classroom Exercise-3



- No two players are from the same country
- Many-to-many?
- Many-to-one?
- One-to-one?

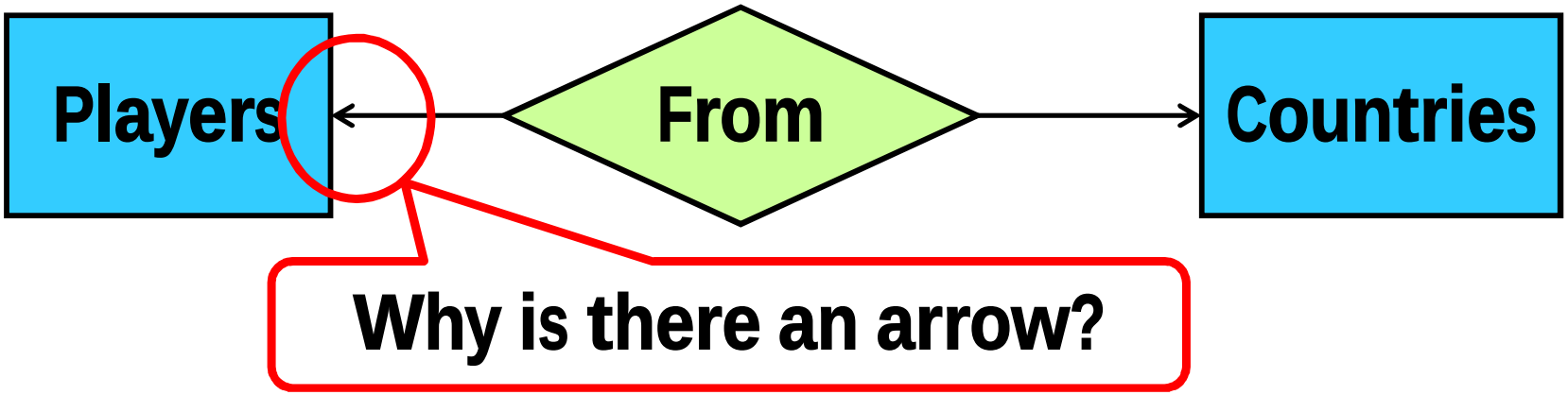
Classroom Exercise-3: Solution



- No two players are from the same country
- Many-to-many?
- Many-to-one?
- One-to-one?

Because we make an assumption: One player cannot be from many countries

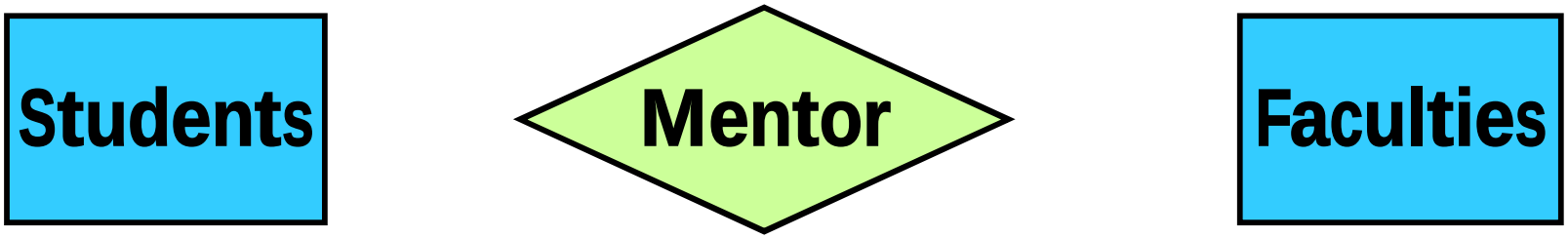
Classroom Exercise-3: Solution



Assumption can be made when it is ambiguous/ unclear in the question without that assumption. However, your assumption must always be a realistic assumption and you must write your assumption together with your answer (e.g., ER diagram).

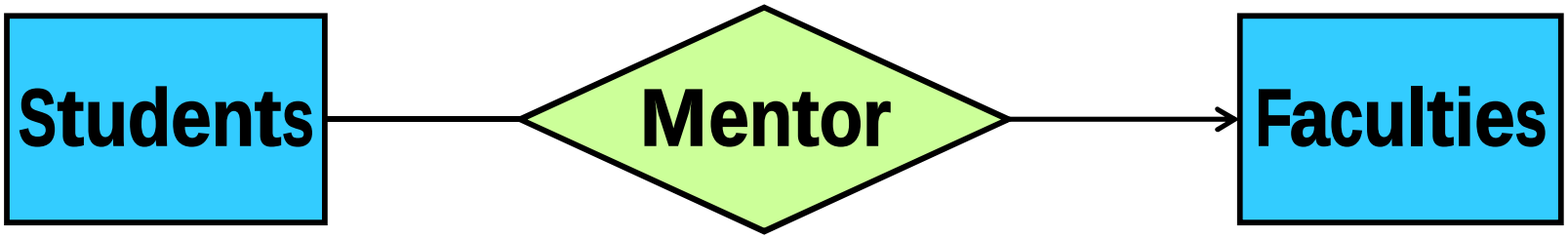
Because we make an assumption: One player cannot be from many countries

Classroom Exercise-4



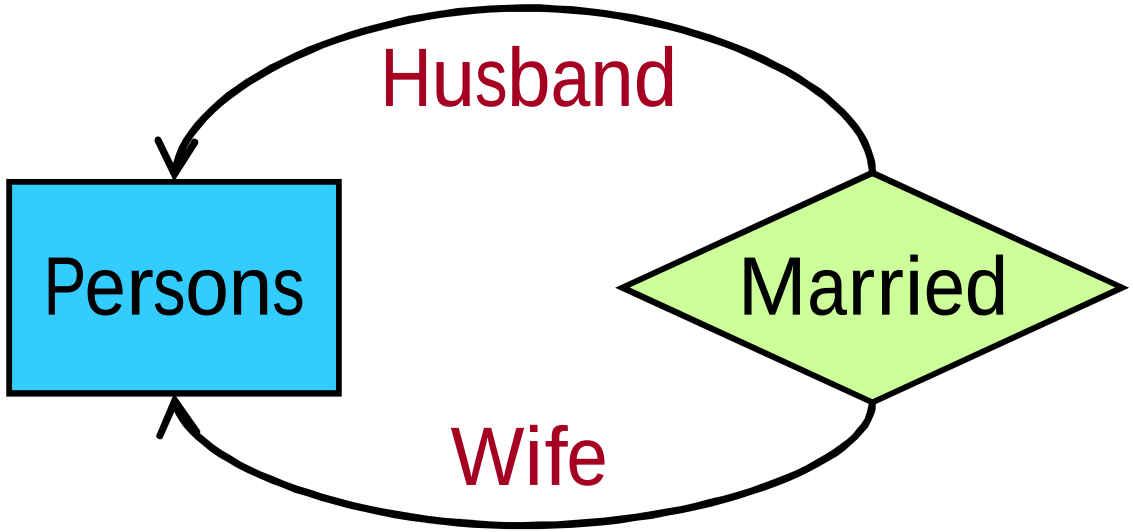
- Each student is mentored by one faculty
- Each faculty can mentor multiple students

Classroom Exercise-4: Solution



- Each student is mentored by one faculty
- Each faculty can mentor multiple students

Roles

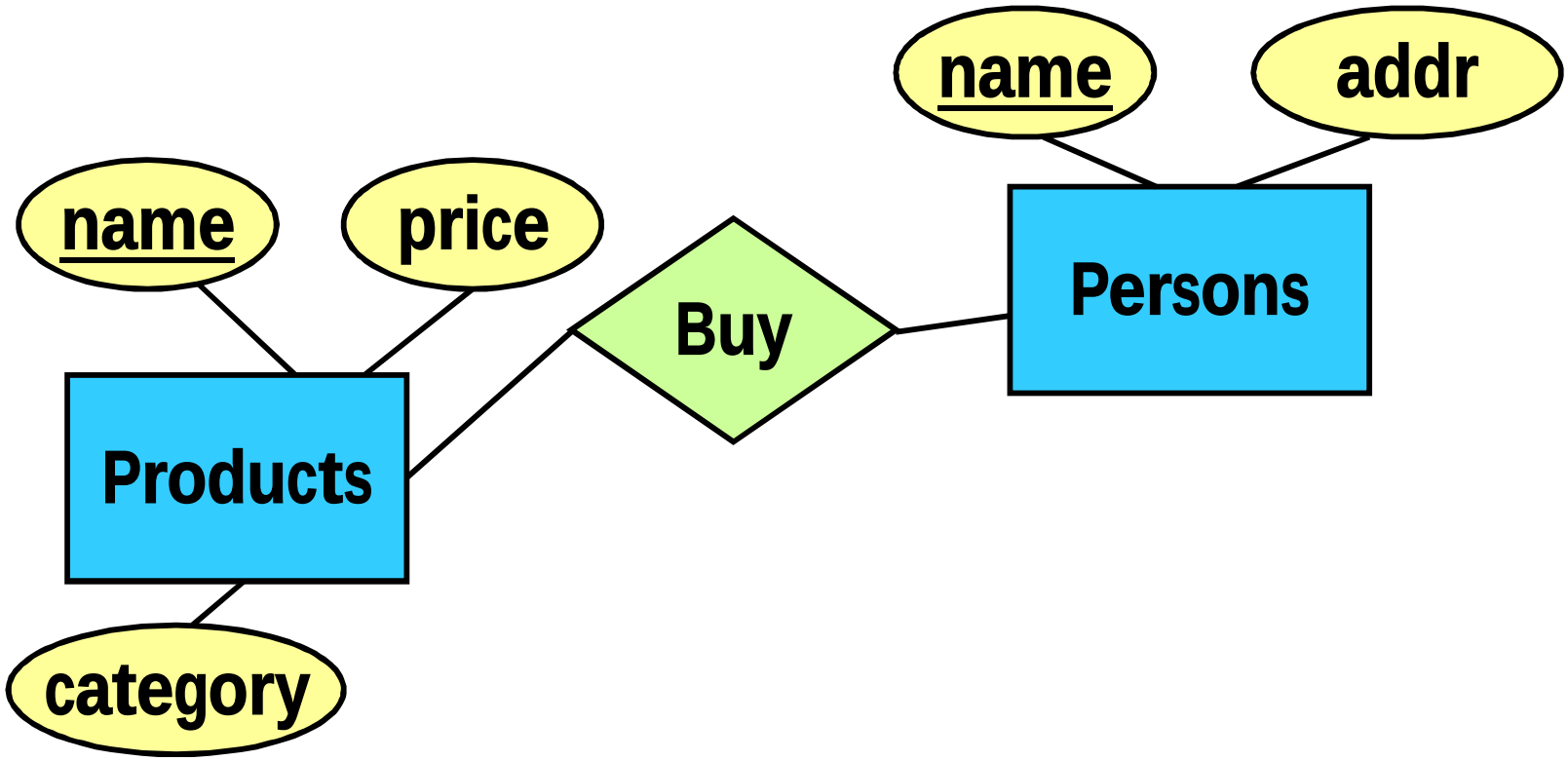


- Sometimes an entity set may appear more than once in a relationship
- Example: some persons are married to each other
- The role of the person is specified on the edge connecting the entity set to the relationship

Husband	Wife
Bob	Alice
David	Cathy
...	...

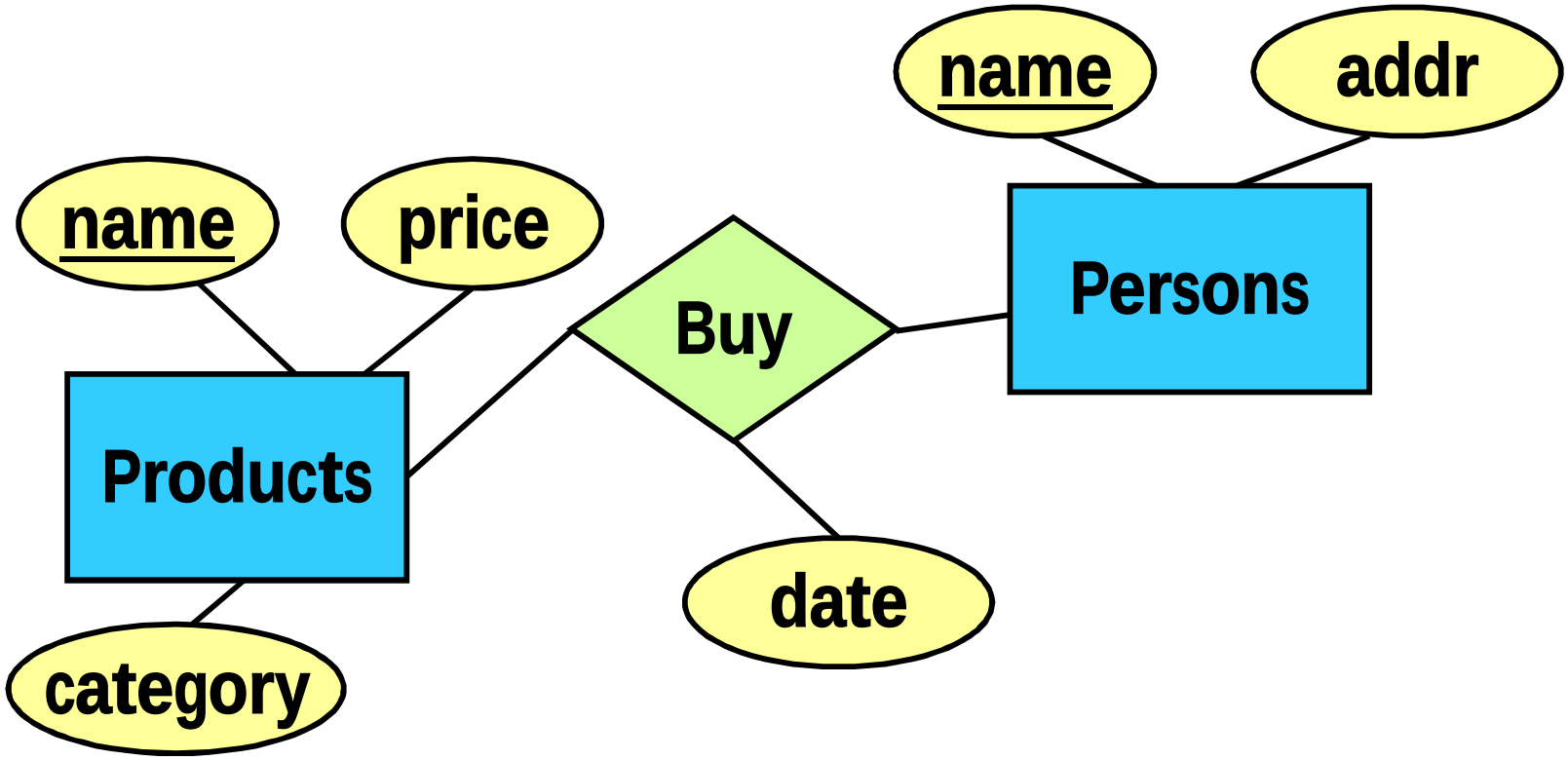
One More Thing about Relationships

- A relationship can have its own attribute



One More Thing about Relationships

- A relationship can have its own attribute
- If we want to record the date of the purchase

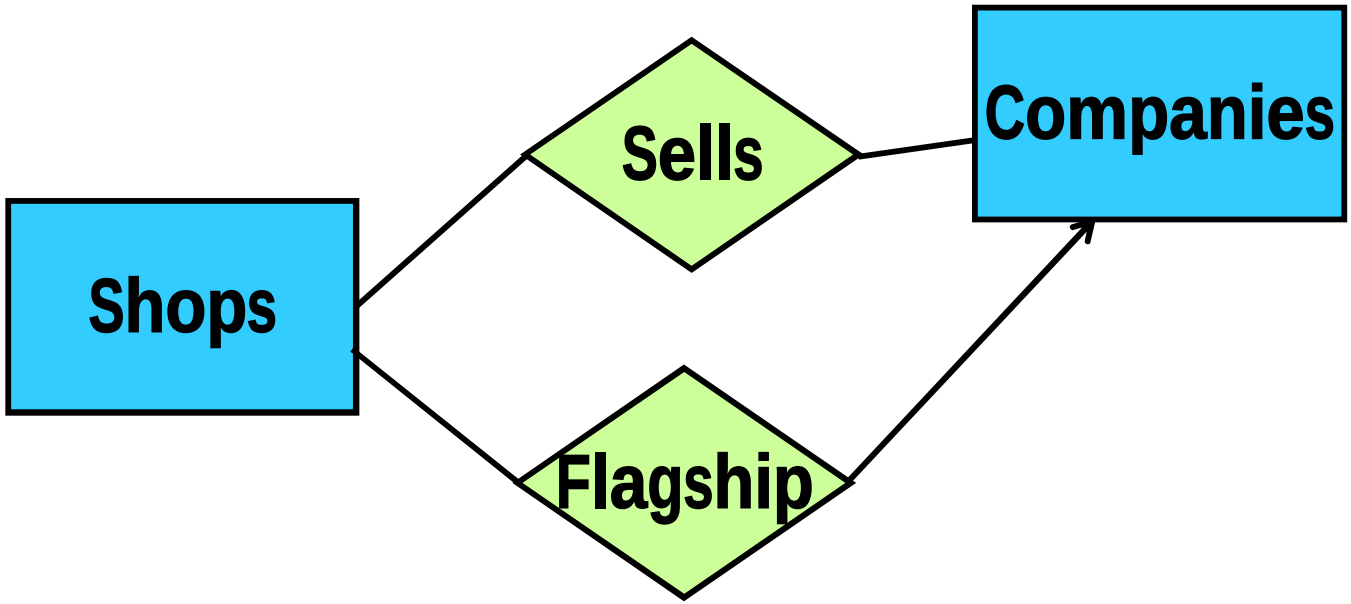


Classroom Exercise-5

- Consider two entity sets, Shops and Companies
- Each shop sells products from at least one company
- Each company has its product sold at least one shop
- A shop may be the flagship shop of at most one company
- Each company has at least one flagship shops
- Draw some relationships between Shops and Companies to capture the above statements

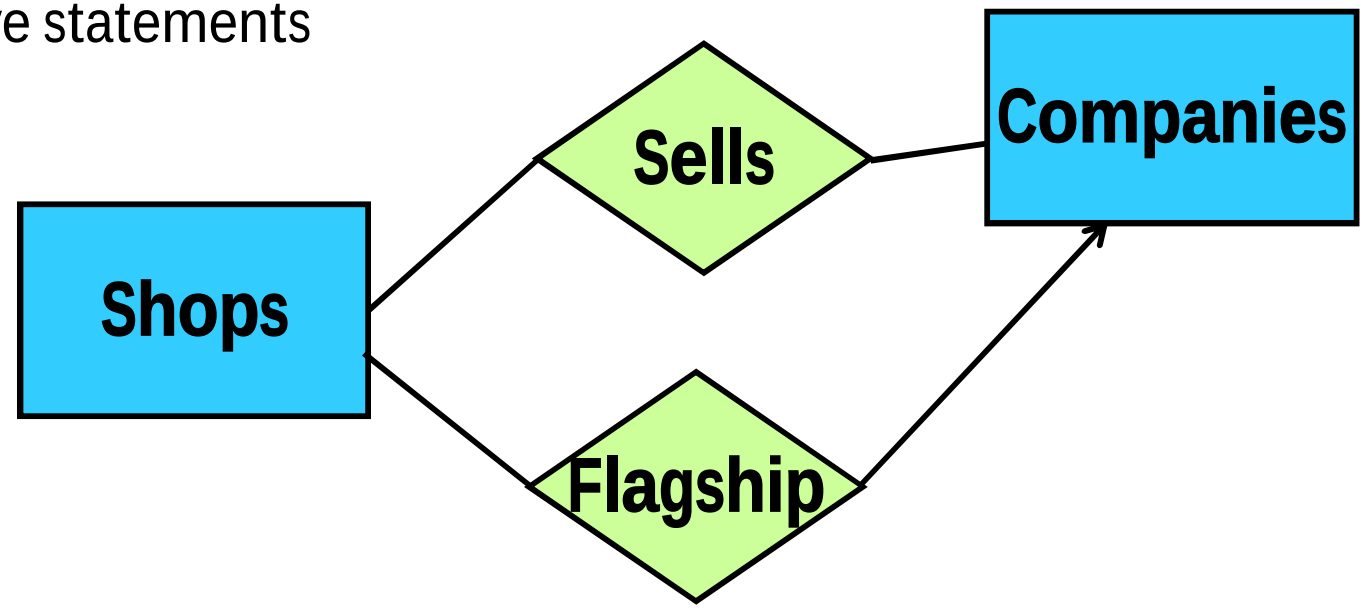
Classroom Exercise-5: Solution

- Consider two entity sets, Shops and Companies
- Each shop sells products from at least one company
- Each company has its product sold at least one shop
- A shop may be the flagship shop of at most one company
- Each company has at least one flagship shops
- Draw some relationships between Shops and Companies to capture the above statements



Classroom Exercise-5: Solution

- Consider two entity sets, Shops and Companies
- Each shop sells products from at least one company
- Each company has its product sold at least one shop
- A shop may be the flagship shop of at most one company
- Each company has at least one flagship shops
- Draw some relationships between Shops and Companies to capture the above statements

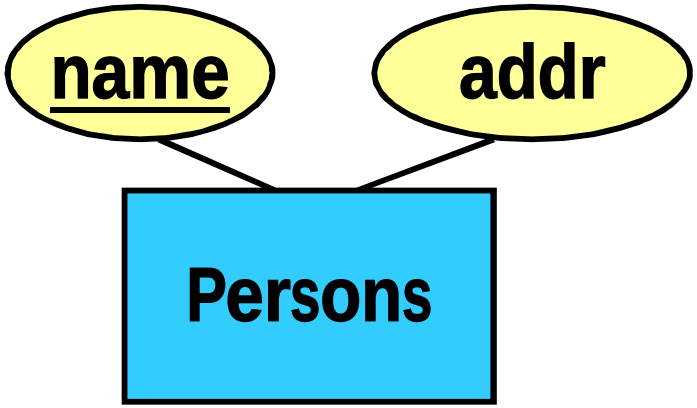


- There can be multiple relationships between two entity sets

Constraints

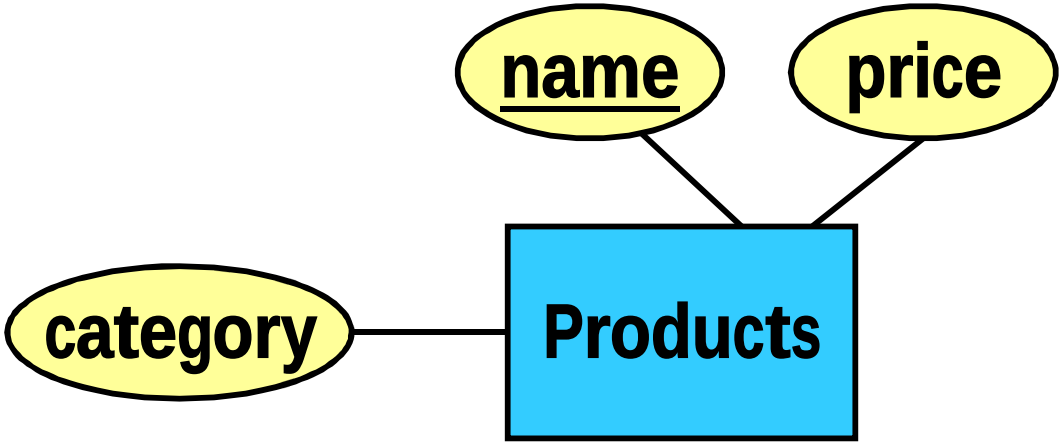
- Some conditions that entity sets and relationships should satisfy
- We will focus on three types of constraints
  - Key constraints
  - Referential integrity constraints
  - Degree constraints

Key



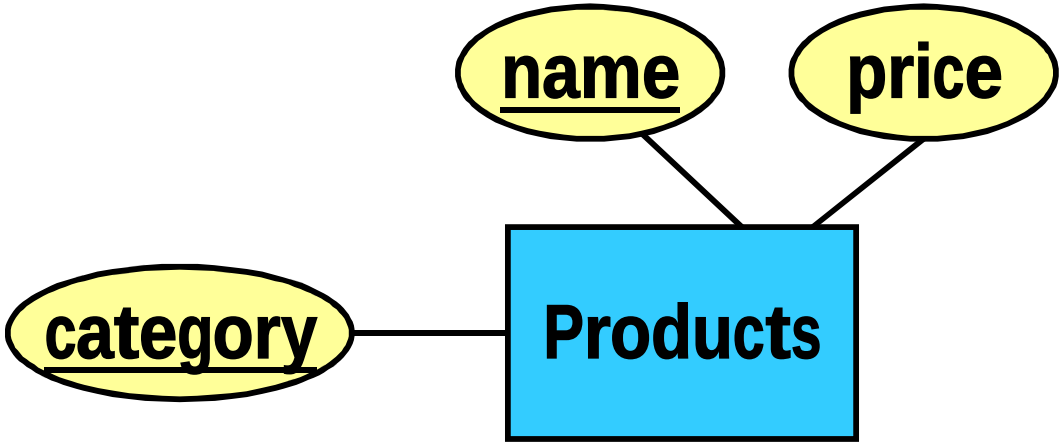
- One or more attributes that are underlined
- Meaning: They uniquely represent each entity in the entity set
- Example: The names uniquely represent the persons
- i.e., each person must have a unique name

Key



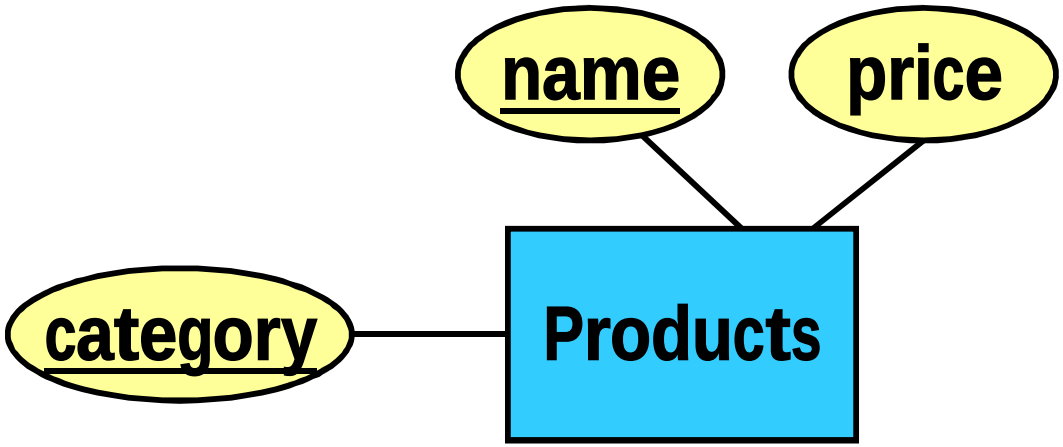
- One or more attributes that are underlined
- Meaning: They uniquely represent each entity in the entity set
- Example: Each product has a unique name

Key



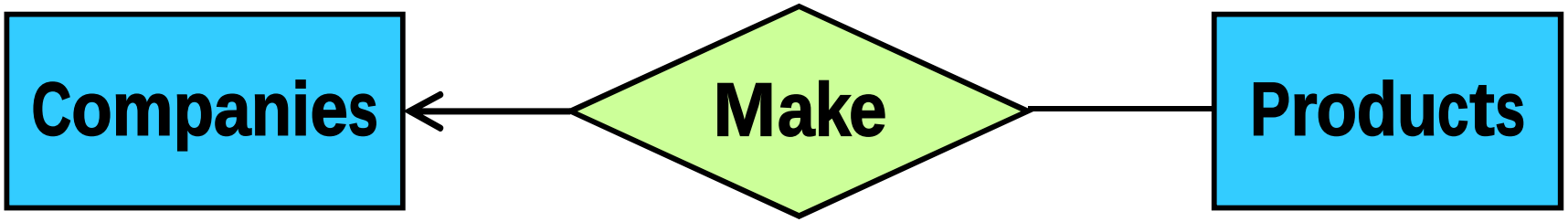
- One or more attributes that are underlined
- What now?
- Each product has a unique <name, category> combination
- But there can be products with the same name, or the same category, but not both
- Example
  - Name = “Apple”, Category = “Fruit”, Price = “1”
  - Name = “Apple”, Category = “Phone”, Price = “888”

Key



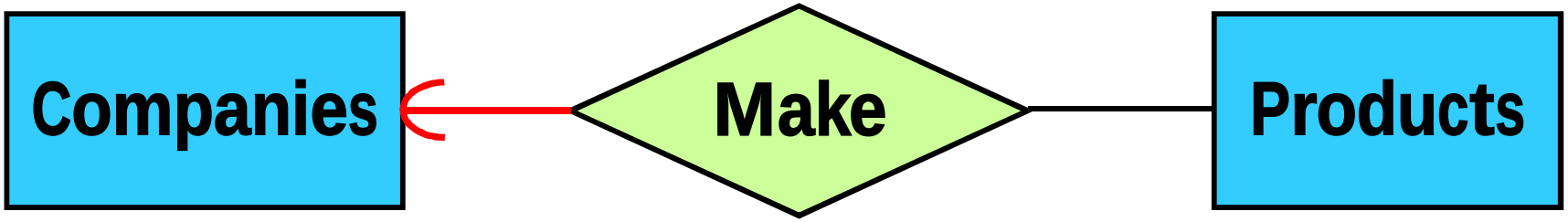
- Rule: Every entity set should have a key
  - So that we can uniquely refer to each entity in the entity set

Referential Integrity



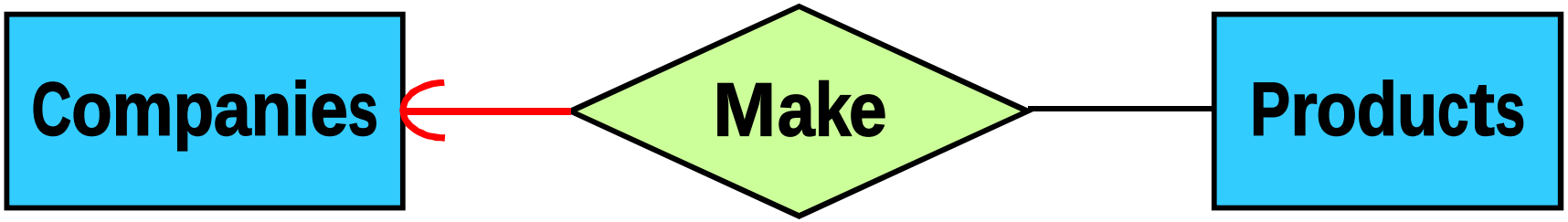
- One company may make multiple products
- One product is made by one company
- Can there be a product that is not made by any company?
- No.
- i.e., every product must be involved in the Make relationship
- This is called a referential integrity constraint.
- How do we specify this in an ER diagram?
- Use a rounded arrow instead of a pointed arrow

Referential Integrity



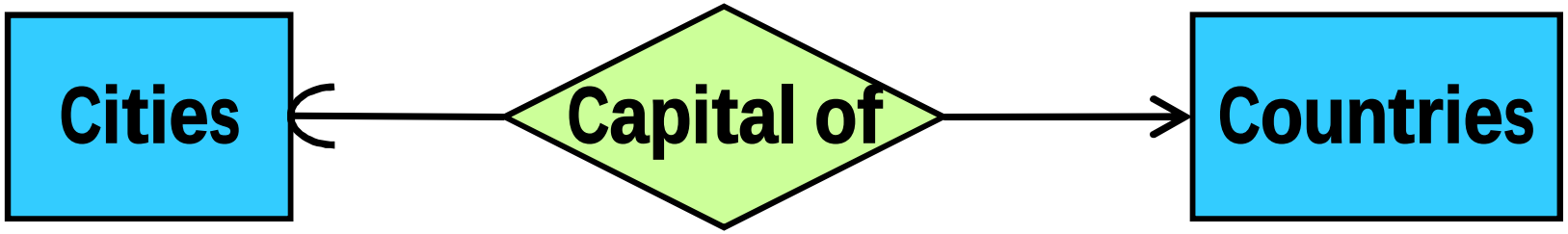
- One company may make multiple products
- One product is made by one company
- Can there be a product that is not made by any company?
- No.
- i.e., every product must be involved in the Make relationship
- This is called a referential integrity constraint.
- How do we specify this in an ER diagram?
- Use a rounded arrow instead of a pointed arrow

Referential Integrity



- What if every company should make at least one product?
- No arrow there….
- In general, a referential integrity constraint can only apply to the “one” side of
  - A many-to-one relationship, or
  - A one-to-one relationship
- Why? Will be clear later in this course
- For the “many” side, there is another type of constraints to use (will be discussed shortly)

Referential Integrity: Classroom Exercise-6



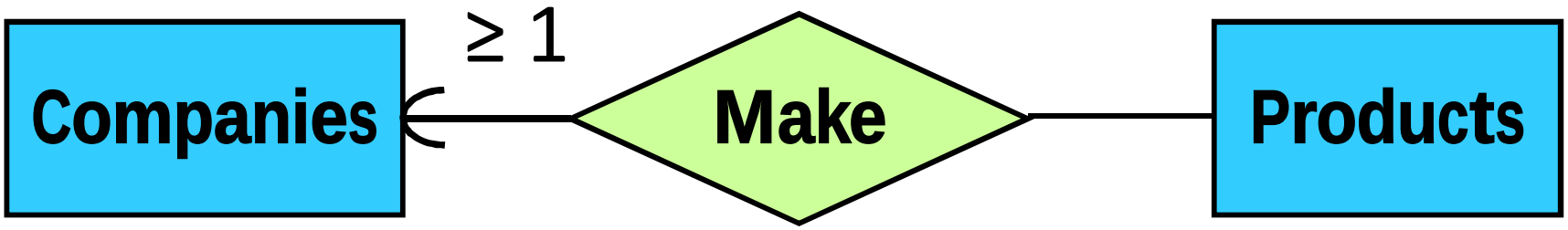
- A city can be the capital of only one country
- A country must have a capital

Referential Integrity: Classroom Exercise-7

- A company must hire at least one person
- A person must be hired by exactly one company

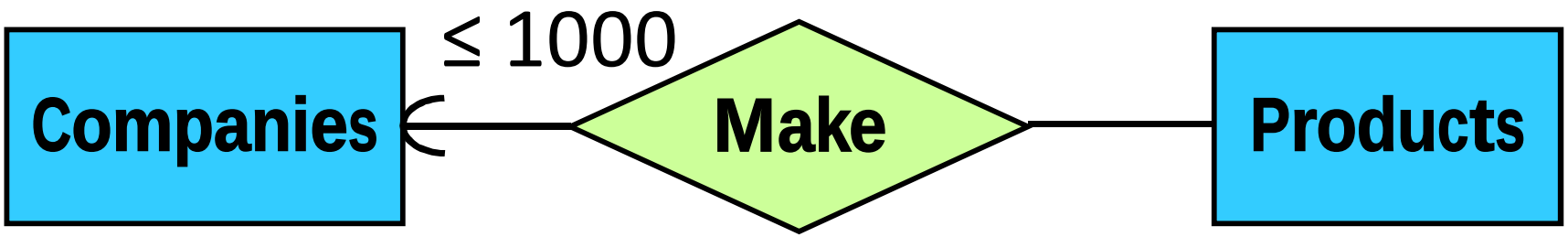
Questions?

Degree Constraint



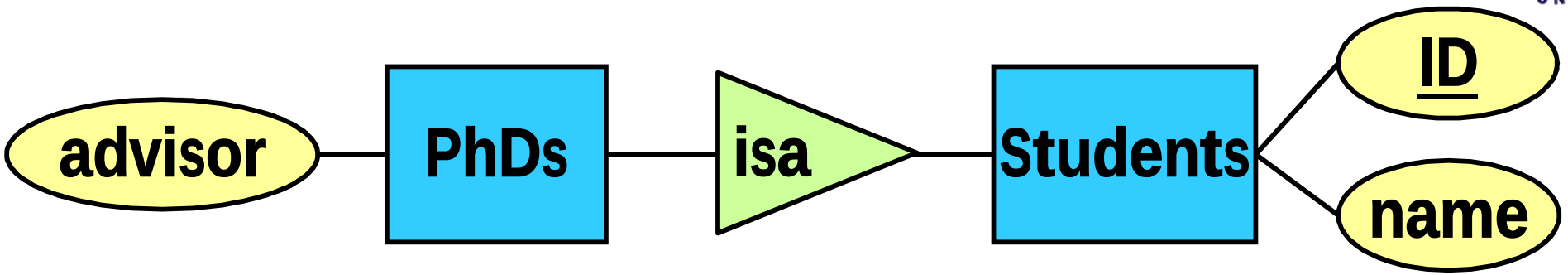
- Each company should make at least 1 product

Degree Constraint



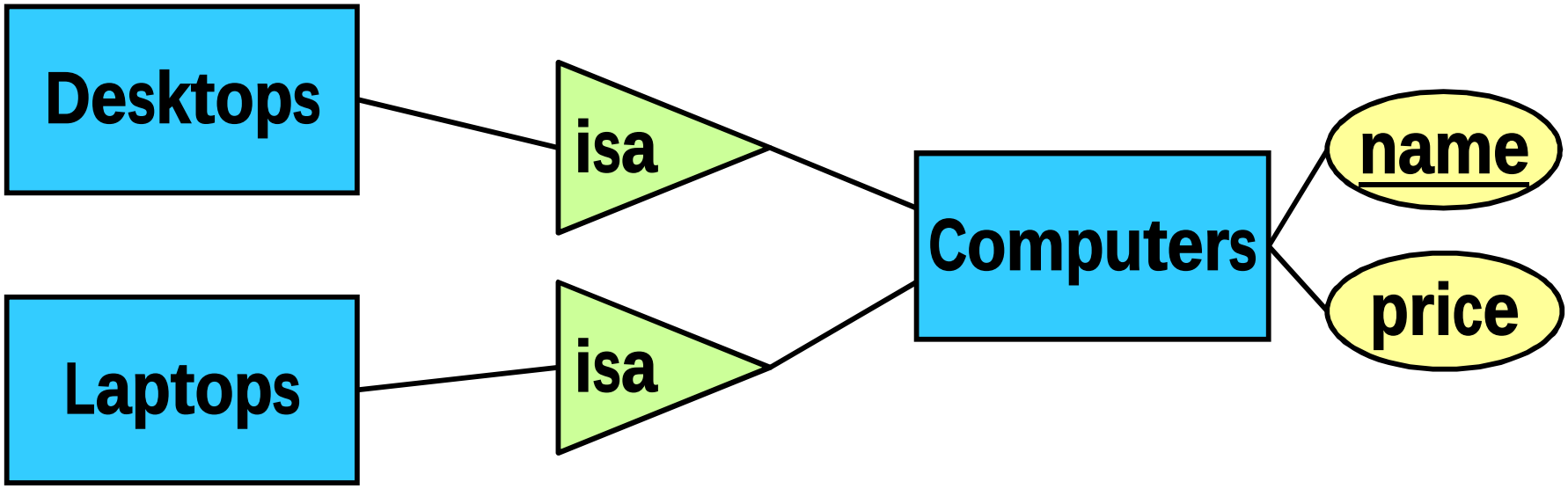
- Each company can make at most 1000 product
- Note
  - Degree constraint not required in the exercise/exam
  - Degree constraints are not easy to enforce in a DBMS
  - Key and referential integrity constraints can be easily enforced

Subclass



- PhDs are a special type of Students
- Subclass = Special type
- The connection between a subclass and its superclass is captured by the isa relationship, which is represented using a triangle
- Key of a subclass = key of its superclass
- Example: Key of Phds = Students.ID
- Students is referred to as the superclass of PhDs

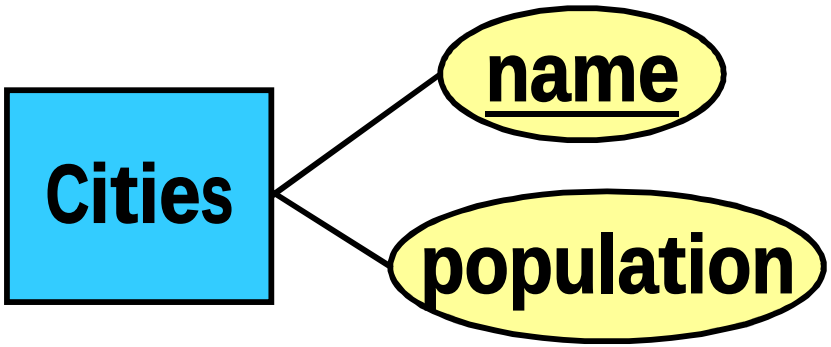
Subclass



- An entity set can have multiple subclasses
- Example
  - Superclass: Computers
  - Subclass 1: Desktop
  - Subclass 2: Laptop

Weak Entity Sets

- Weak entity sets are a special type of entity sets that
  - cannot be uniquely identified by their own attributes
  - needs attributes from other entities to identify themselves
- Example: Cities in USA
- Problem: there are cities with identical names



WIKIPEDIA
The Free Encyclopedia

Main page
Contents
Featured content
Current events
Random article
Donate to Wikipedia

Interaction
Help
About Wikipedia
Community portal
Recent changes
Contact Wikipedia

Toolbox

Print/export

Languages
Български
Dansk
Deutsch
Español
Français
한국어
Italiano
עברית
Kiswahili
Magyar
Nederlands
日本語
Norsk (bokmål)
Polski
Português
Română
Русский
Slovenčina
Suomi
Svenska
Volapük

Madison

From Wikipedia, the free encyclopedia

Madison may refer to:

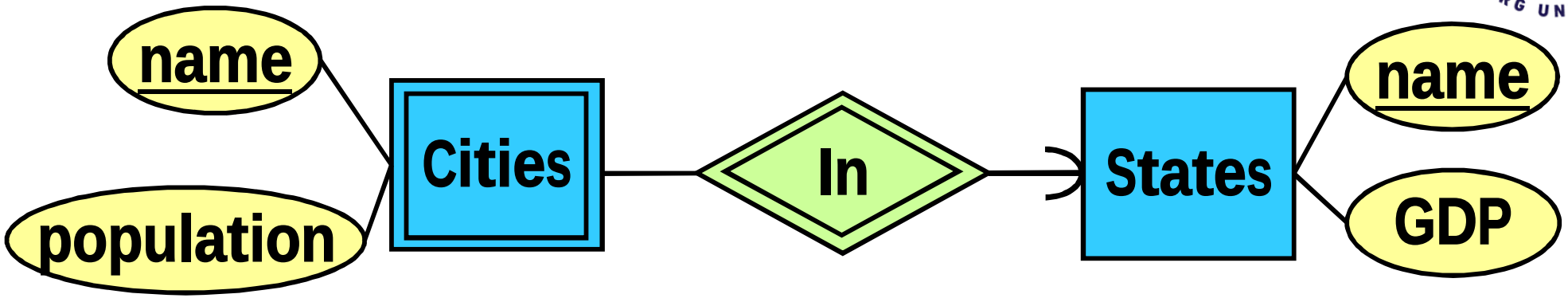
People

- Madison (name), a given name and a surname

Place names

- Madison, Wisconsin, the largest city by the name and the state capital of Wisconsin
- Madison, Alabama
- Madison, Arkansas
- Madison, California
- Madison, Connecticut
- Madison, Florida
- Madison, Georgia
- Madison, Illinois
- Madison, Indiana
- Madison, Kansas
- Madison, Maine
  - Madison (CDP), Maine, census-designated place within the town of Madison
- Madison, Minnesota
- Madison, Mississippi
- Madison, Missouri
- Madison, Nebraska
- Madison, New Hampshire
- Madison, New Jersey
- Madison (town), New York
  - Madison (village), New York, within the town of Madison
- Madison, North Carolina
- Madison, Ohio
- Madison, Pennsylvania
- Madison, South Dakota
- Madison, Tennessee
- Madison, Virginia
- Madison, West Virginia
- Madison (town), Wisconsin, adjacent to the city of Madison
- Madison Lake, Minnesota
- Madison Park, Seattle, Washington State

Weak Entity Sets



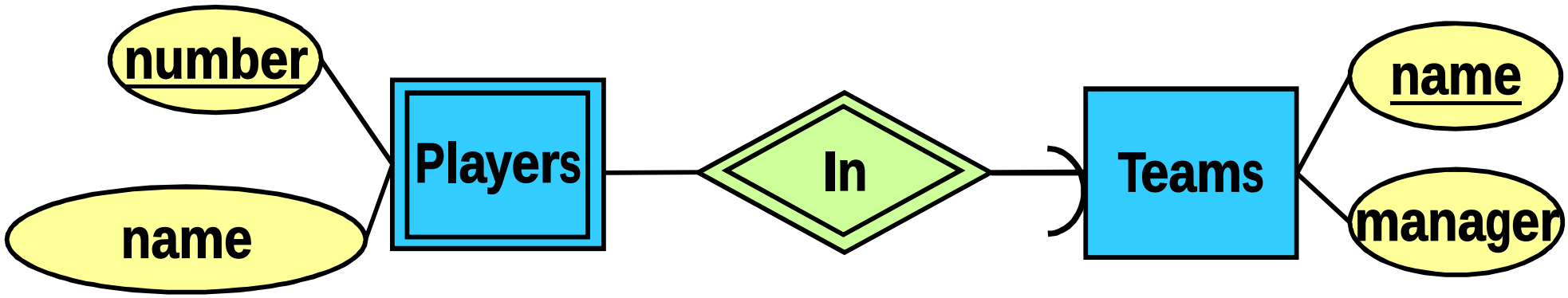
- Problem: there are cities with identical names
- Observation: cities in the same state would have different names
- Solution: make Cities a weak entity set associated with the entity set States
- The relationship In is called the supporting relationship of Cities
- Weak entity set = Double-lined rectangle
- Supporting relationship = Double-lined diamond
- The key of Cities = (State.name, Cities.name)

Classroom Exercise-8

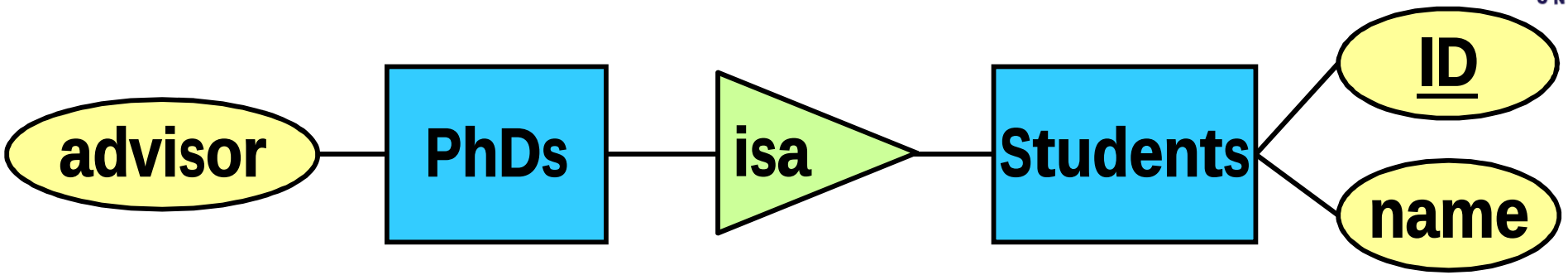
- Consider two entity sets: Players and Teams
- Each player has a name and a number
- Each team has a name and a manager
- Each player plays for exactly one team, and is uniquely identified within the team by his/her number
- Each team is uniquely identified by its name
- Different players may have the same name
- Draw a ER diagram that captures the above statements
- What is the key of Players?

Classroom Exercise-8: Solution

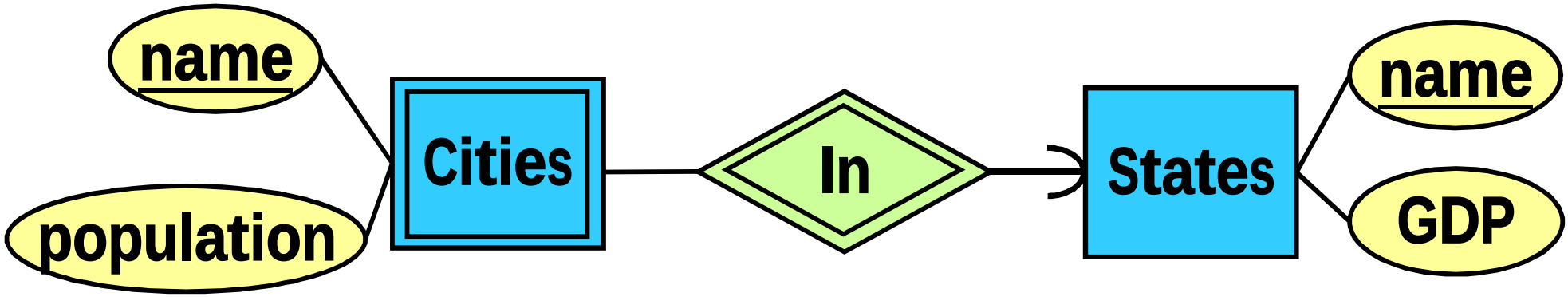
- Consider two entity sets: Players and Teams
- Each player has a name and a number
- Each team has a name and a manager
- Each player plays for exactly one team, and is uniquely identified within the team by his/her number
- Each team is uniquely identified by its name
- Different players may have the same name
- Draw a ER diagram that captures the above statements
- What is the key of Players? (Players.number, Teams.name)



Subclass vs. Weak Entity Sets



- PhDs are a special type of Students



- Cities are NOT a special type of States

Questions?

Summary

- Introduction 
- Entity-relationship diagram 

Study-at-Home Slides

ER Design Principles

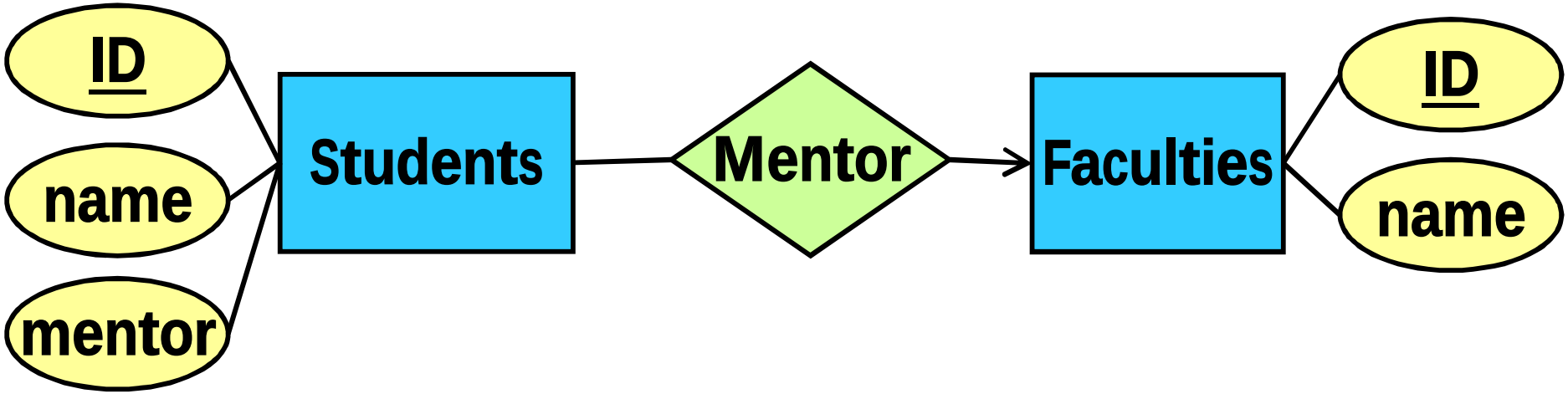
Will be in the syllabus of Exercise, self-study, and Exam

Design Principle 1: Be Faithful

- Be faithful to the specifications of the application
- Capture the requirements as much as possible

Design Principle 2: Avoid Redundancy

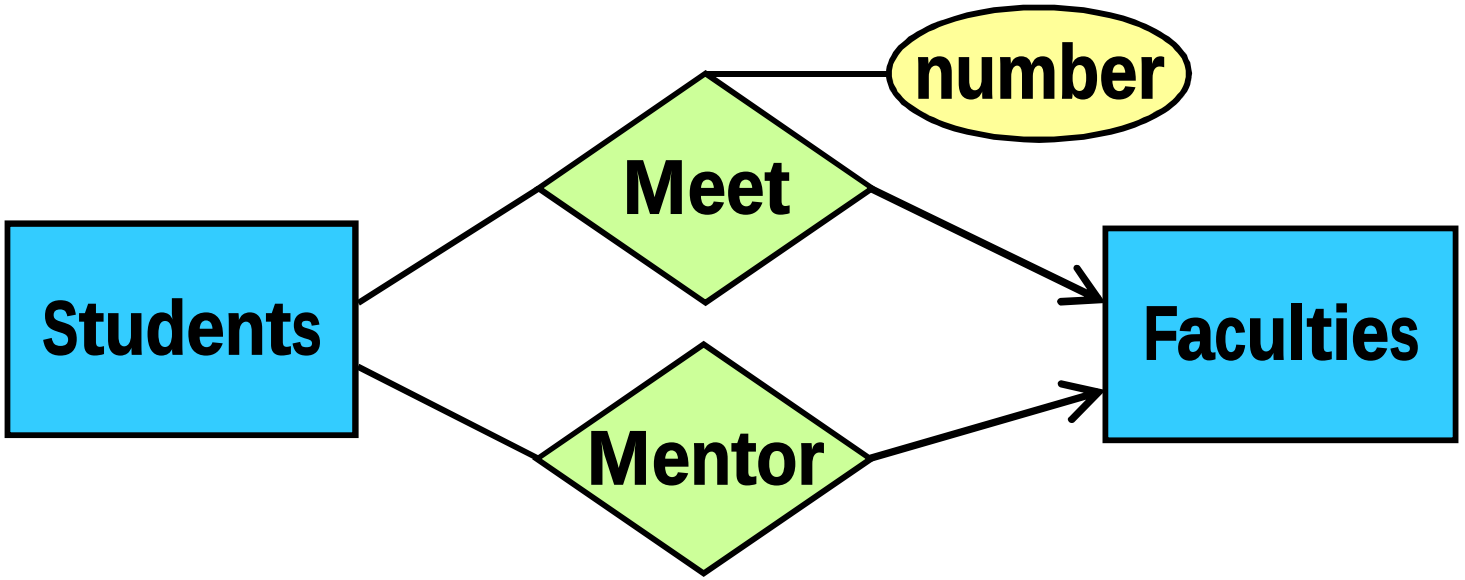
- Avoid repetition of information
- Example



- Problems that can be caused by redundancy
  - Waste of space
  - Possible inconsistency

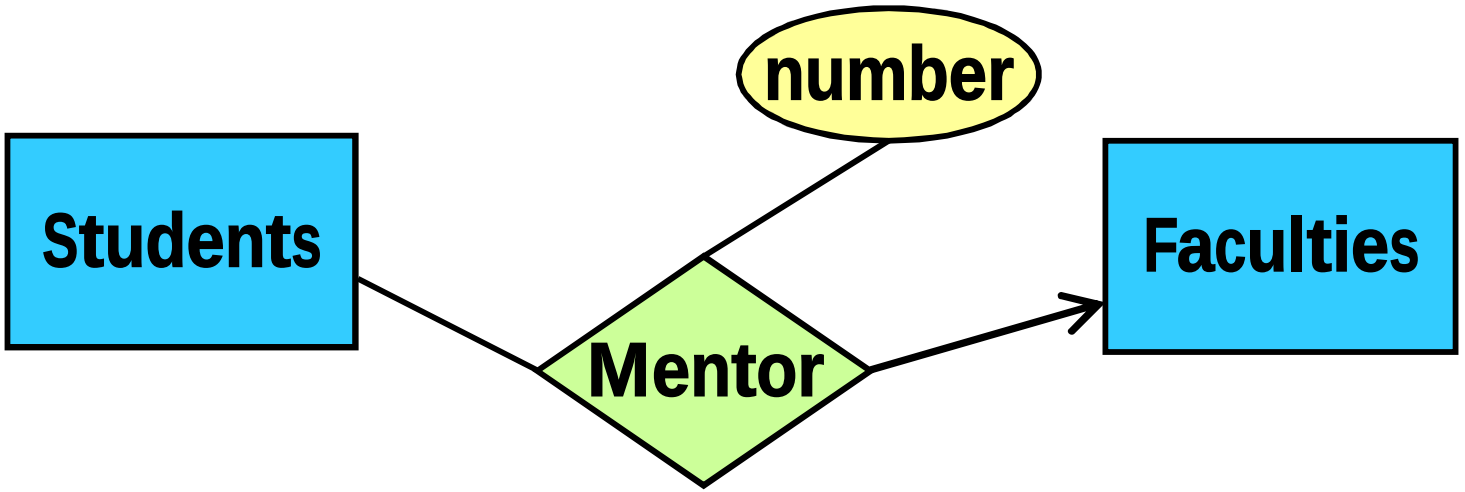
Design Principle 3: Keep It Simple

- Each student is mentored by one faculty
- One faculty can mentor multiple students
- We also record the number of times that a mentee meets with his/her mentor
- Design below: Not wrong, but can be simplified



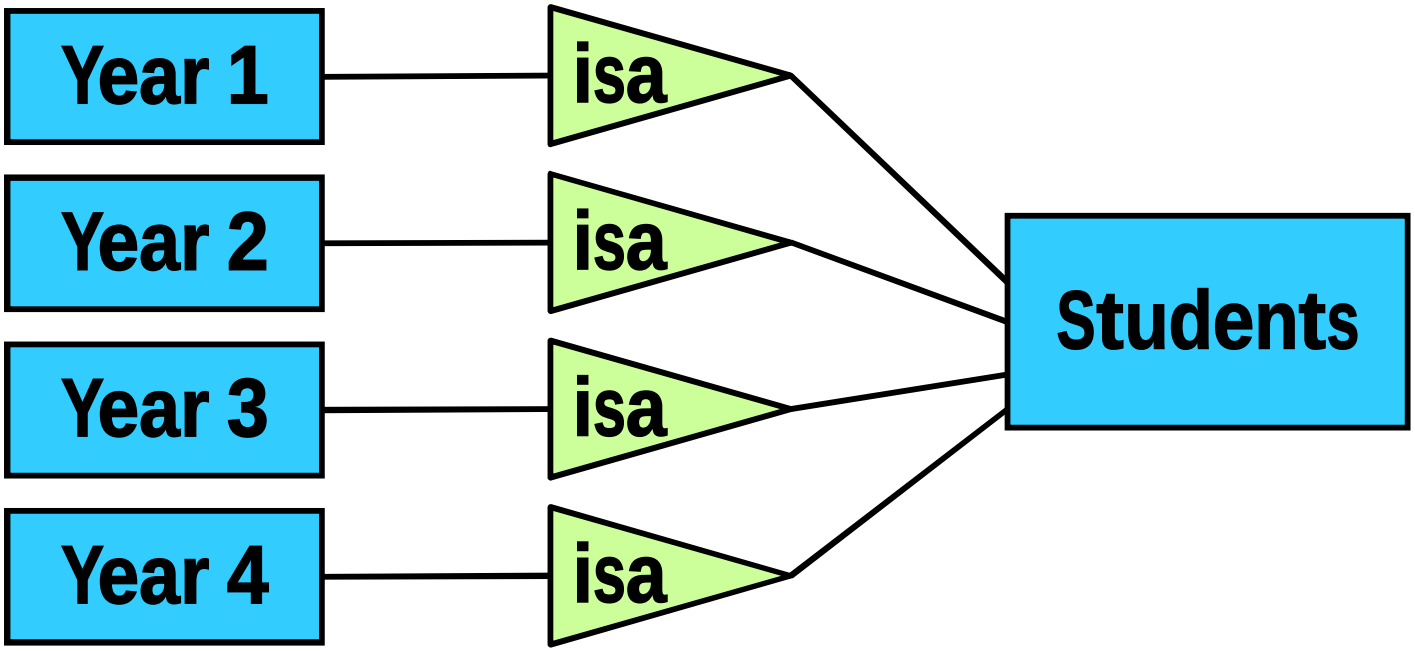
Design Principle 3: Keep It Simple

- Each student is mentored by one faculty
- One faculty can mentor multiple students
- We also record the number of times that a mentee meets with his/her mentor
- Better Design:



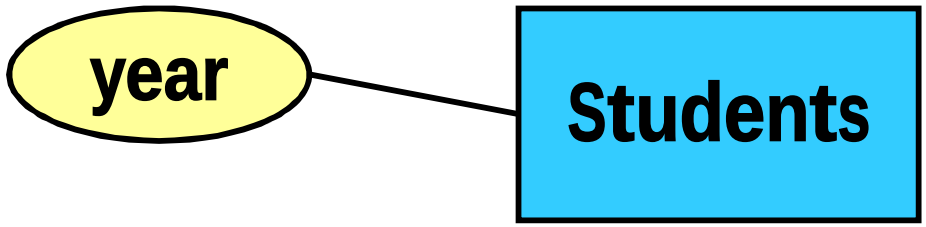
Design Principle 3: Keep It Simple

- There are four types of students: Year 1, Year 2, Year 3, Year 4
- Design below: Not wrong, but can be simplified



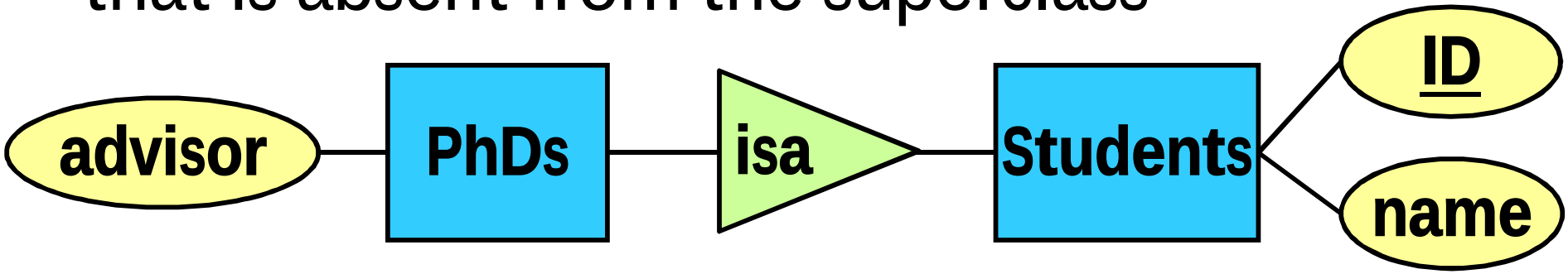
Design Principle 3: Keep It Simple

- There are four types of students: Year 1, Year 2, Year 3, Year 4
- Better Design

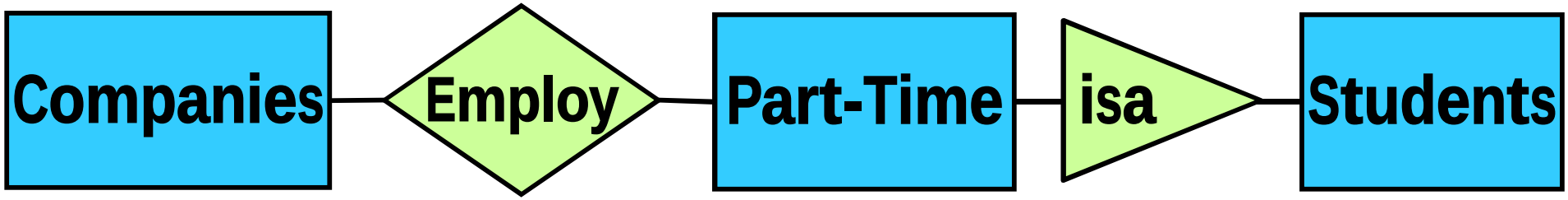


Tips: When to Use Subclasses

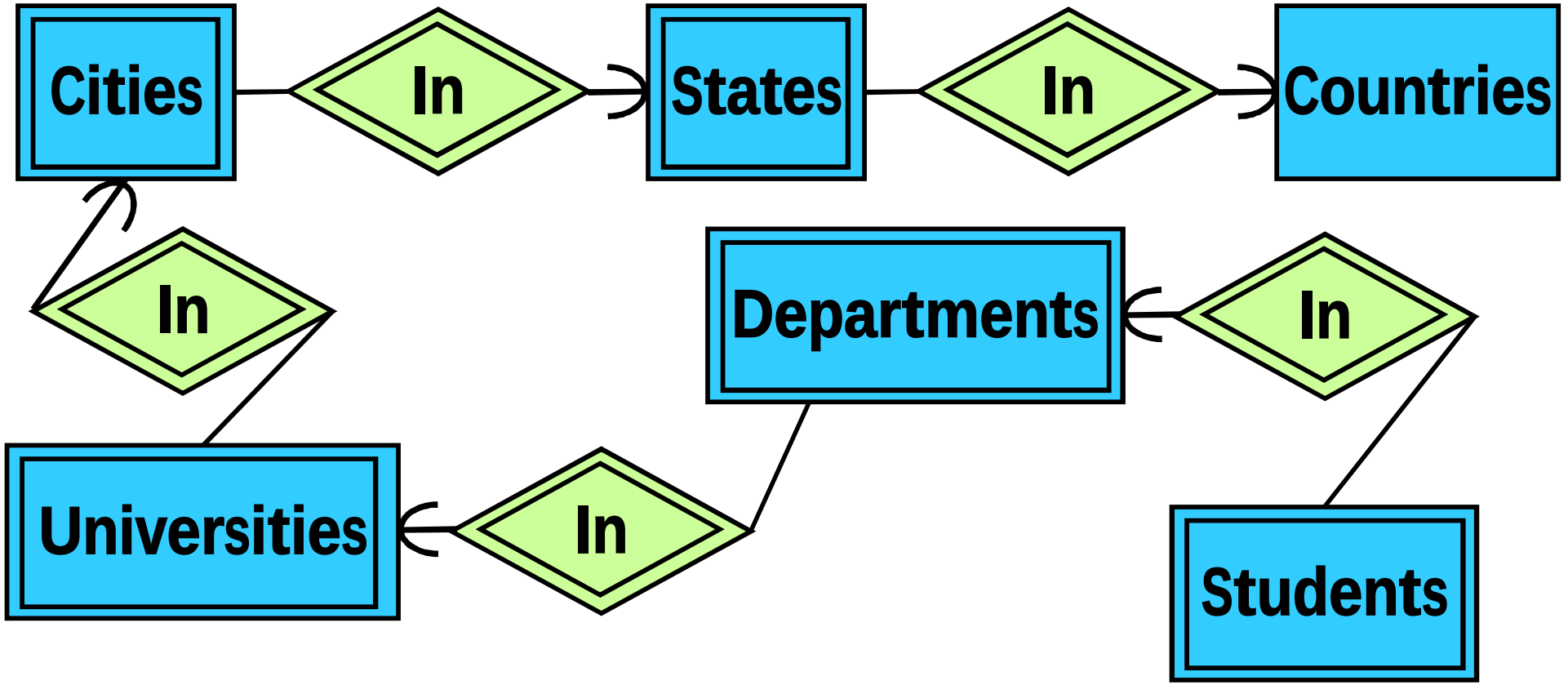
- Case 1: When a subclass has some attribute that is absent from the superclass



- Case 2: When a subclass has its own relationship with some other entity sets



Design Principle 4: Don't Over-use Weak Entity Sets



- Too many entity sets that should not be "weak"

Questions ??

Thank You !