Data Modelling

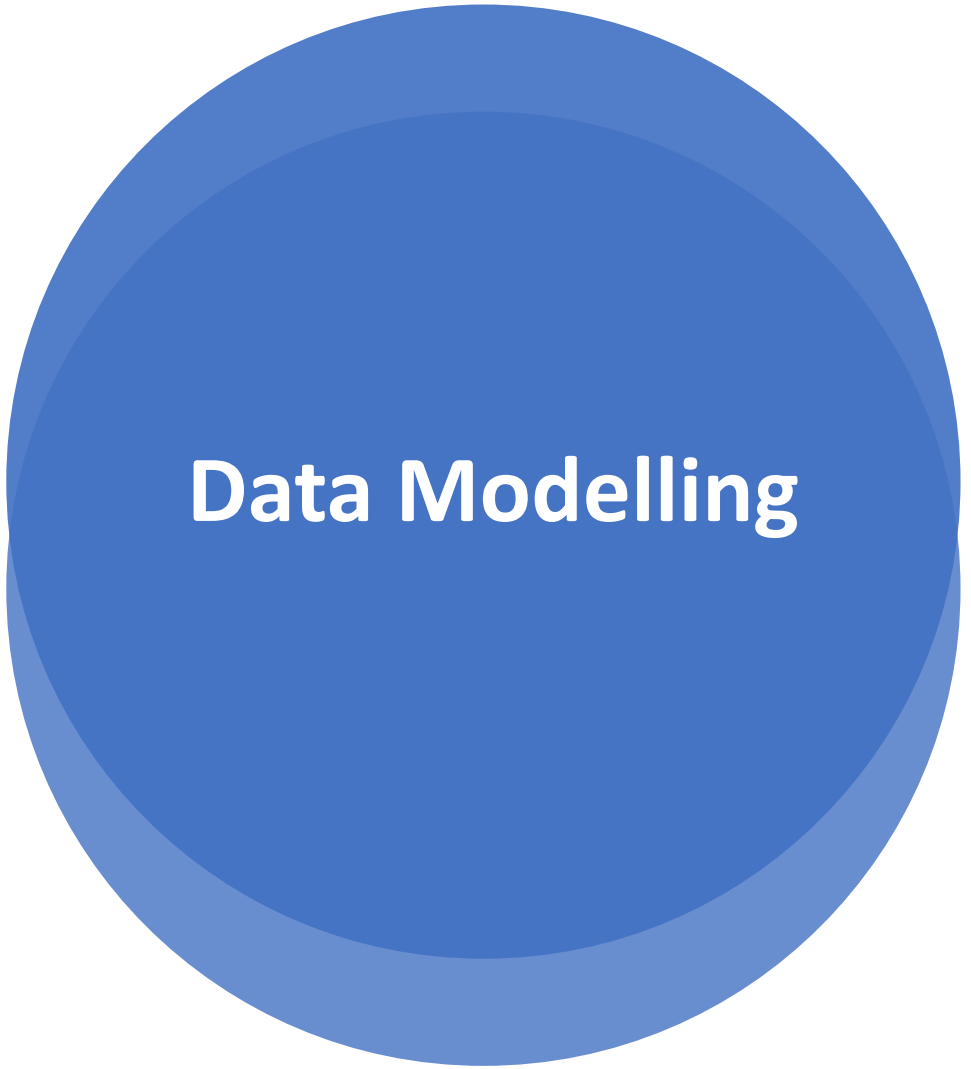
SSTC 2022
Module 3 – Lecture 2

Dr. Salvatore Flavio Pileggi

SalvatoreFlavio.Pileggi@uts.edu.au
https://www.uts.edu.au/staff/salvatoreflavio.pileggi

School of Computer Science, Faculty of Engineering and IT
University of Technology Sydney (Australia)

- Data Requirements
- Data Modelling
- Entity-Relationship (ER) Model
- Dealing with complexity (FAQs)



Data Requirements & Data Modelling

Data Modelling

Data Requirements

What is a “data requirement”?

- Informally, it’s a specification of **requirements from a data perspective** – i.e. the identification, prioritisation, formulation, and validation of the data needed to achieve business objectives.

How can analyse and understand the data requirements of an organisation?

- **Data Modelling**

How can we model data?

- Looking at the body of knowledge in the field, several possible approaches
- **ER Model** is probably the most intuitive and effective and, indeed, we adopt it in this subject

Data Modelling

What is it for?

A Data Model:

- identifies the **information** that the organisation will need to hold
- identifies the data that must be **captured, stored and retrieved**
- focuses on **what data is required and how it should be organised.**
- **describes** the data in a **logical (conceptual) manner**
- often relies on graphical representations
- can be approached at a **different level of abstraction**
- normally leads towards a **database design**

ER Model

Entity-Relationship Diagram (ERD)

Why ERD?

- ERD as a “Data Model” is
  - an **abstraction of the data** the organisation works with
  - a way to organise the data of interest into a **standardised structure**

- ERD is a **conceptual data model** that views the real world as consisting of **entities and relationships**

- It may be understood as **graphical representation of the data requirements** for a database including the entities relevant to a chosen problem domain, the relationships between them, and their attributes.

- ERD is a **great asset** at an educational level

Why E-R model?

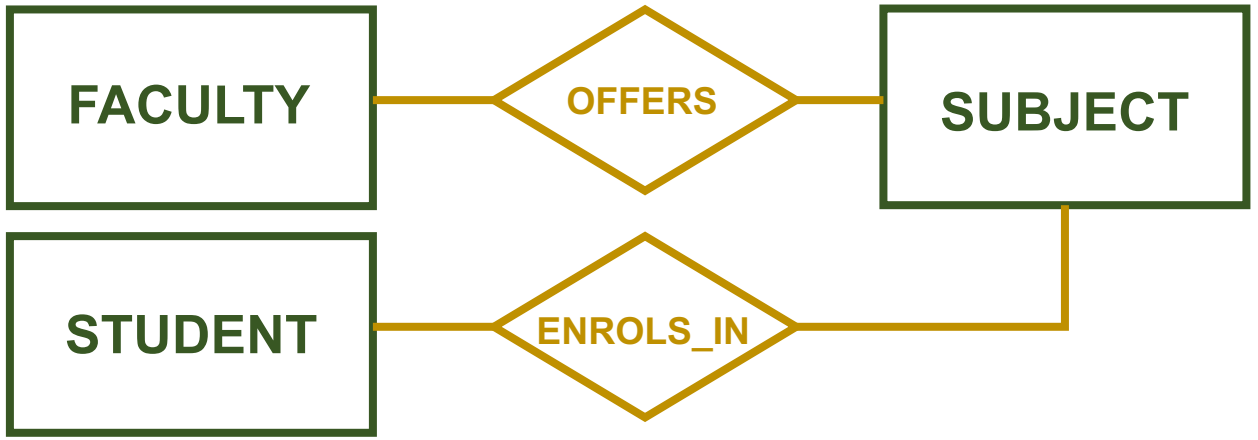
- E-R diagram can express the overall logical structure of a database graphically
- E-R diagrams are simple and clear

Entity-Relationship Diagram (ERD)

Entities & Relationships

As the name suggests, an ERD diagram is composed of two main building blocks, **entities** & **relationships**

If we do understand the nature of these building block and the difference between them, then we can master ERD



*Intuitively, **Faculties offer Subjects** and **Students enrolls in Subjects***

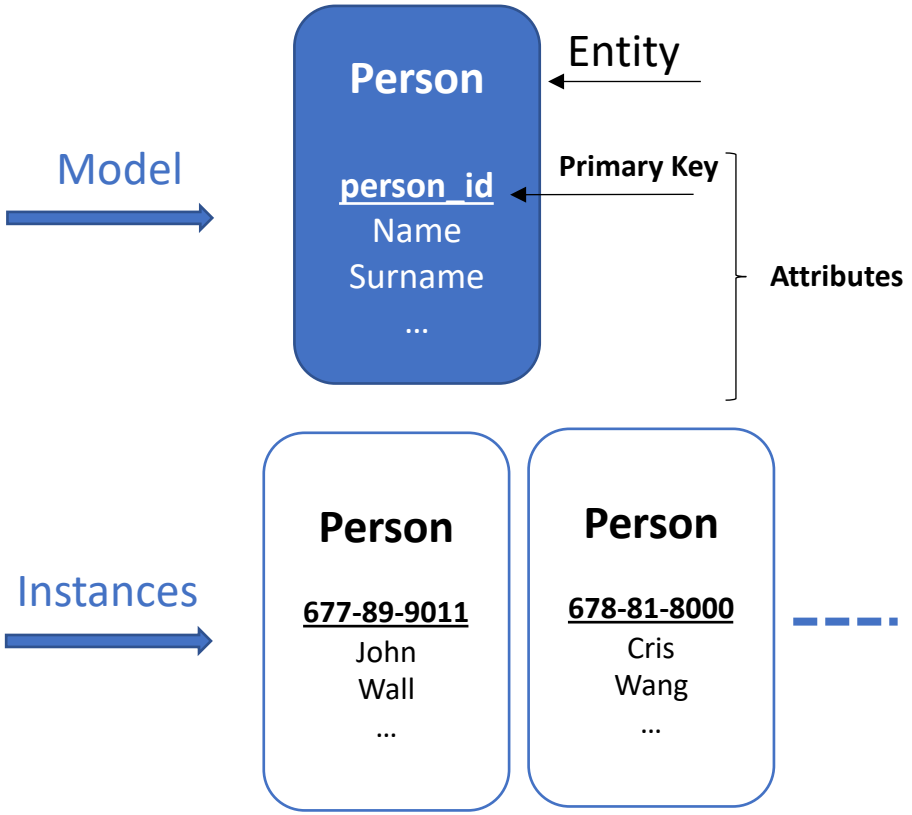
It's a data structure!

Entity-Relationship Diagram (ERD)

Entities

- An **entity** is a **“thing” or “object” in the real world** that is distinguishable from all other objects. For example, each person in a university is an entity.
- An entity has a set of **properties (or attributes)**
- The values for some set of properties must **uniquely identify an entity**. This “special” property (or set of property) is referred to as **Primary Key**.
- An entity may be **concrete**, such a person or a book, or it may be **abstract**, such as a course, a course offering, or a flight reservation.

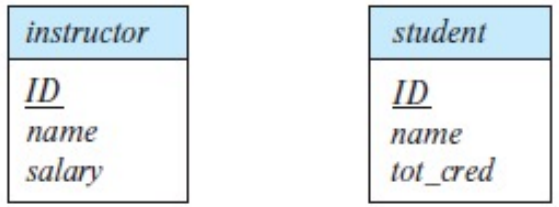
For instance, a person may have a *person_id* property whose value uniquely identifies that person. Thus, the value 677-89-9011 for *person_id* would uniquely identify one particular person in the university.



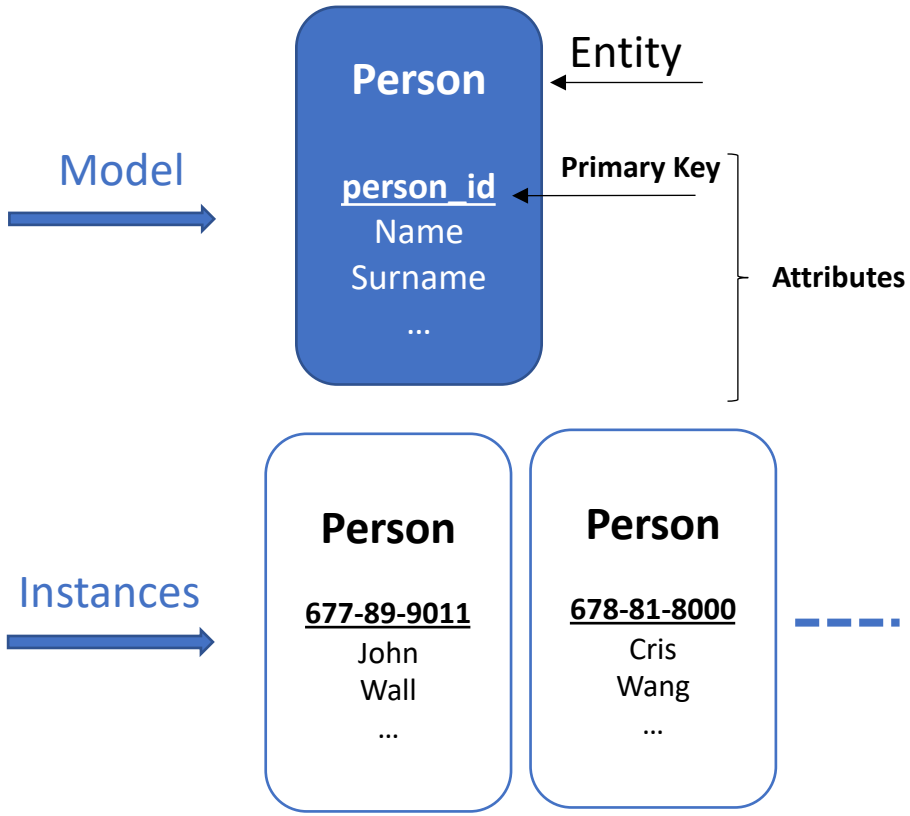
Entity-Relationship Diagram (ERD)

Entities

- An entity set is represented in an E-R diagram by a **rectangle**, which is divided into two parts. The first part, which in this text is shaded blue, contains the **name of the entity set**. The second part contains the names of all the **attributes** of the entity set.
- Attributes that are part of the **primary key** are underlined.



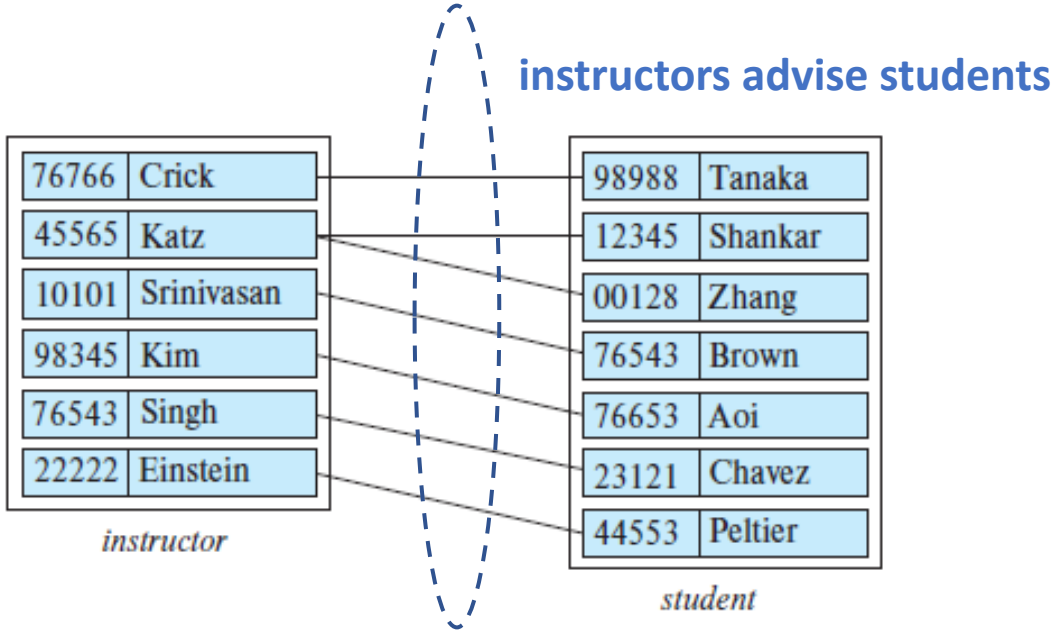
For instance, a person may have a *person_id* property whose value uniquely identifies that person. Thus, the value 677-89-9011 for *person_id* would uniquely identify one particular person in the university.



Entity-Relationship Diagram (ERD)

Relationship

A **relationship** is an association among entities

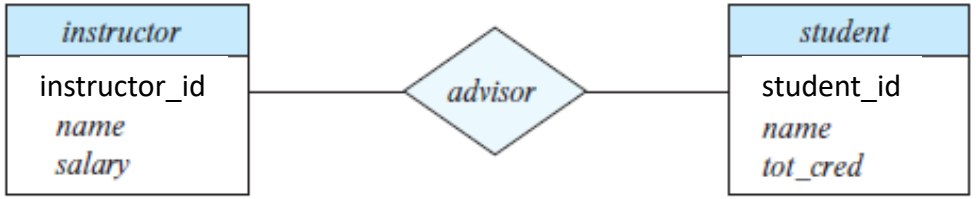


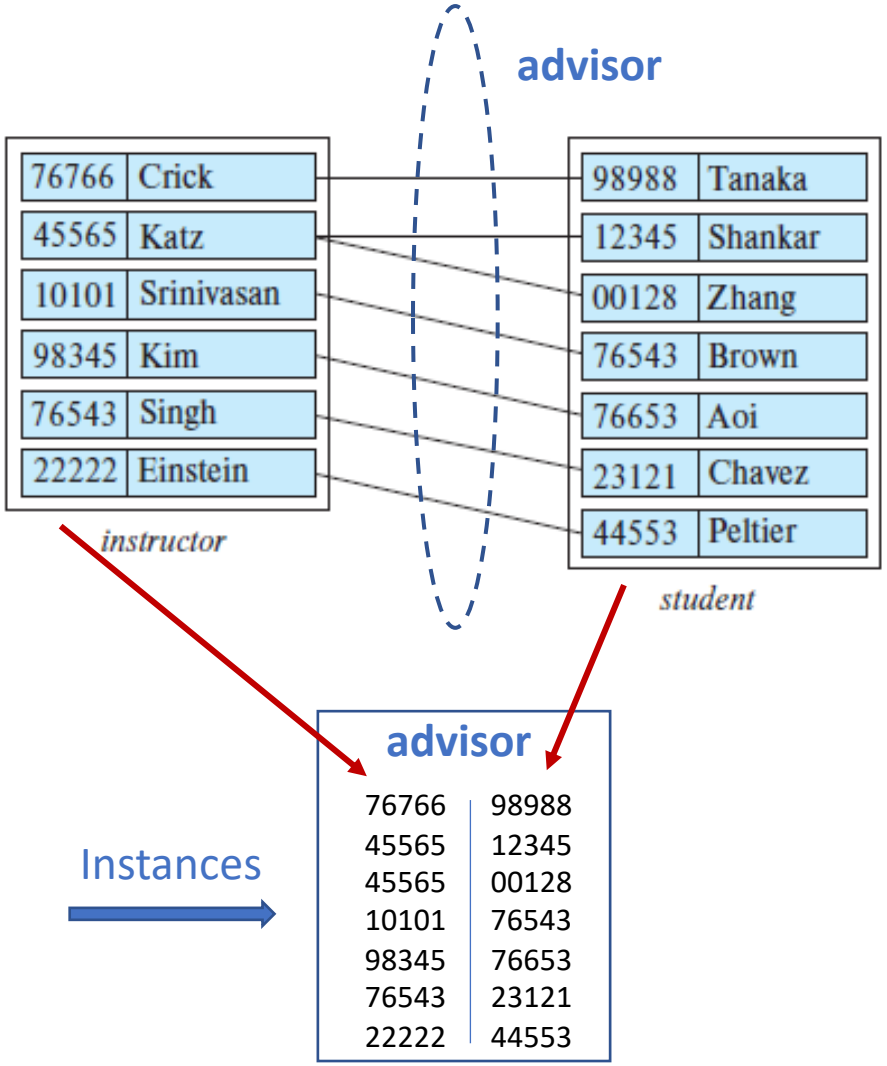
The relationship **"advisor"** is an association between instructors and students

Entity-Relationship Diagram (ERD)

Relationship

A relationship set is represented in an E-R diagram by a **diamond**, which is linked via **lines** to a number of different entity sets (rectangles).

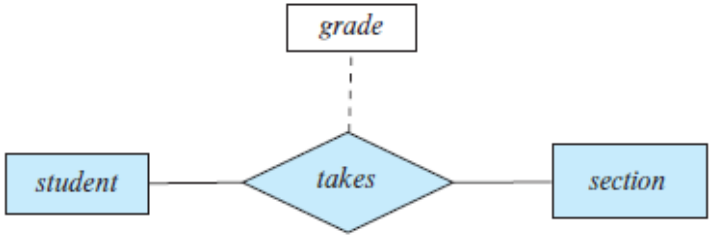




Entity-Relationship Diagram (ERD)

Relationship

Also a relationship may have attributes, called **descriptive attributes**.



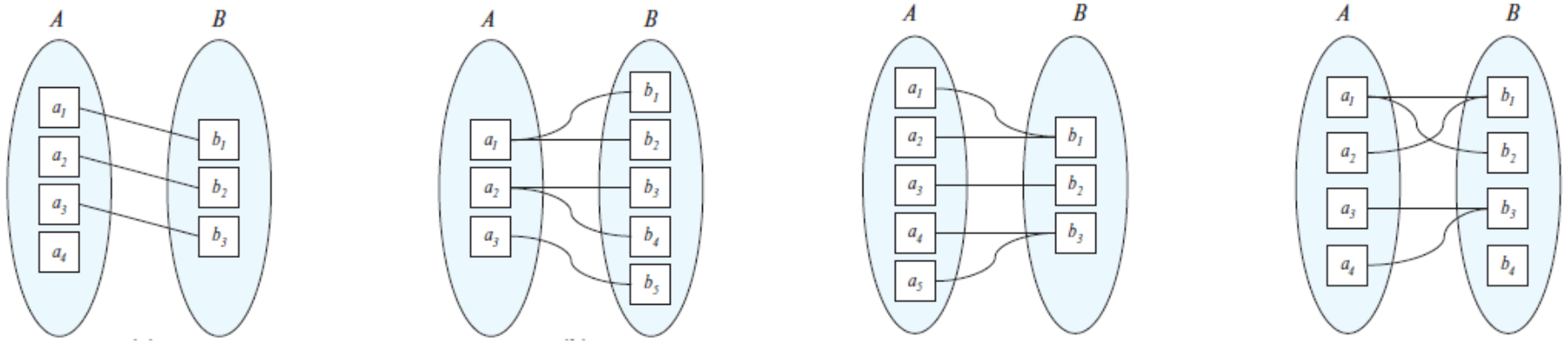
A relationship set may have **multiple descriptive attributes**.

Entity-Relationship Diagram (ERD)

Mapping cardinality

Mapping cardinalities, or cardinality ratios, express the **number of entities to which another entity can be associated via a relationship set**.

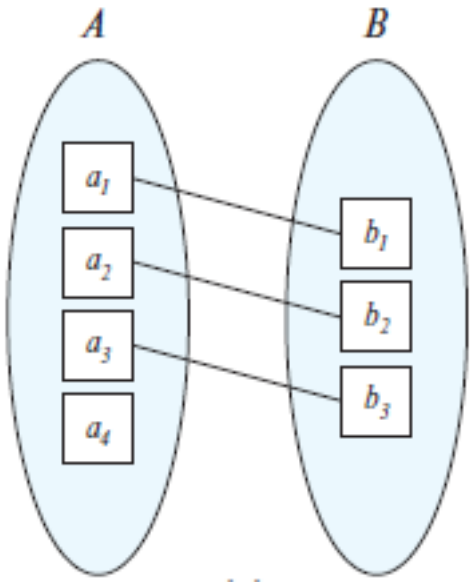
Mapping cardinalities are most useful in describing binary relationship sets, although they can contribute to the description of relationship sets that involve more than two entity sets.

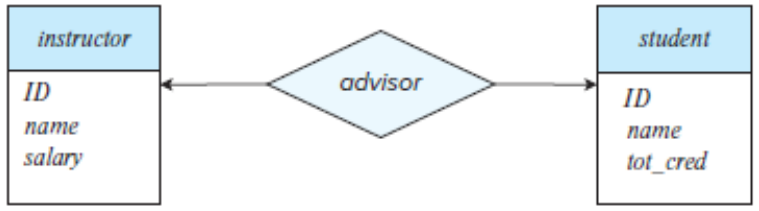


Entity-Relationship Diagram (ERD)

Mapping cardinality: one-to-one relationship

One-to-one. An entity in *A* is associated with *at most* one entity in *B*, and an entity in *B* is associated with *at most* one entity in *A*.





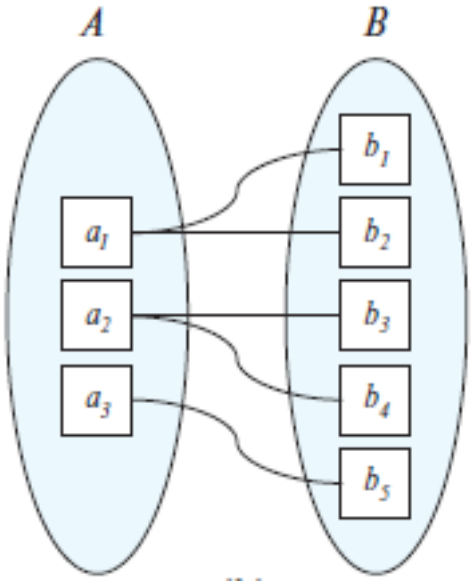
(a) One-to-one

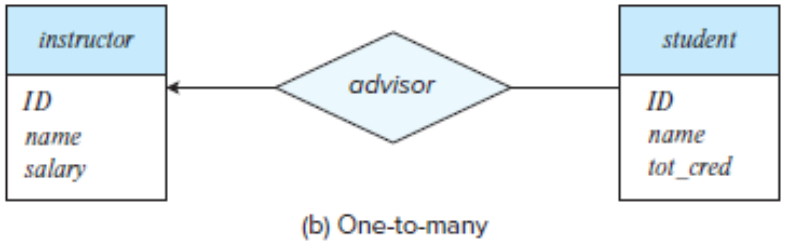
An instructor may advise at most one student, and a student may have at most one advisor.

Entity-Relationship Diagram (ERD)

Mapping cardinality: one-to-many relationship

One-to-many. An entity in *A* is associated with any number (zero or more) of entities in *B*. An entity in *B*, however, can be associated with *at most* one entity in *A*.





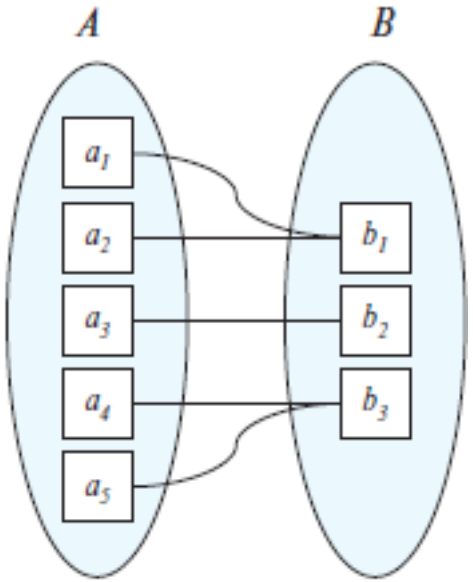
(b) One-to-many

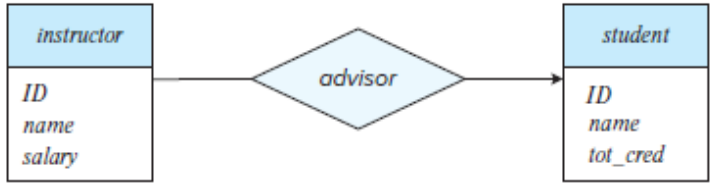
An instructor may advise many students, but a student may have at most one advisor.

Entity-Relationship Diagram (ERD)

Mapping cardinality: many-to-one relationship

Many-to-one. An entity in *A* is associated with *at most* one entity in *B*. An entity in *B*, however, can be associated with any number (zero or more) of entities in *A*.





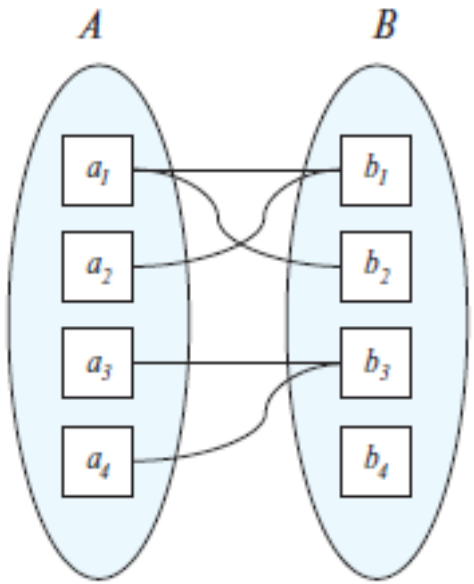
(c) Many-to-one

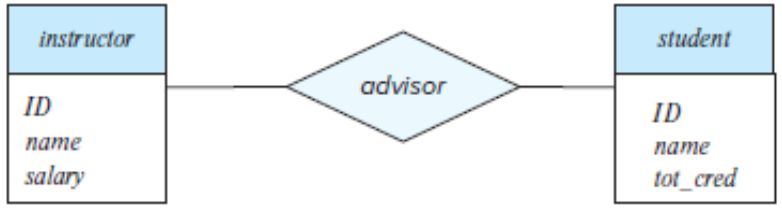
An instructor may advise at most one student, but a student may have many advisors.

Entity-Relationship Diagram (ERD)

Mapping cardinality: many-to-many relationship

Many-to-many. An entity in *A* is associated with any number (zero or more) of entities in *B*, and an entity in *B* is associated with any number (zero or more) of entities in *A*.





(d) Many-to-many

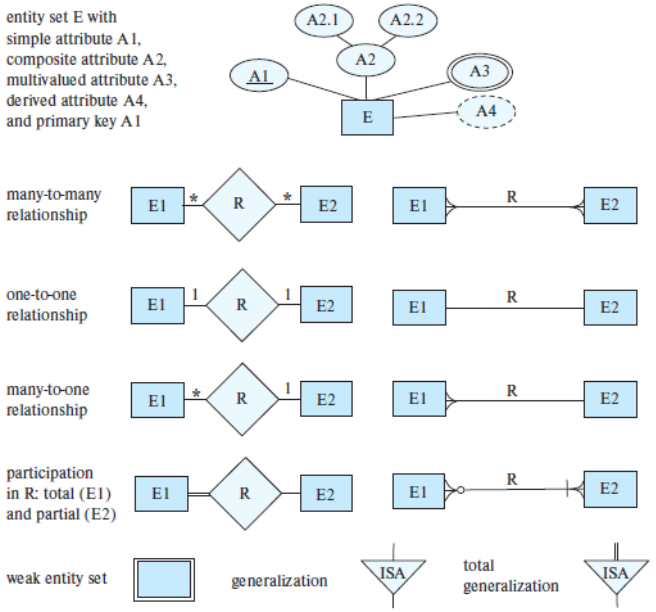
An instructor may advise many students, and a student may have many advisors.

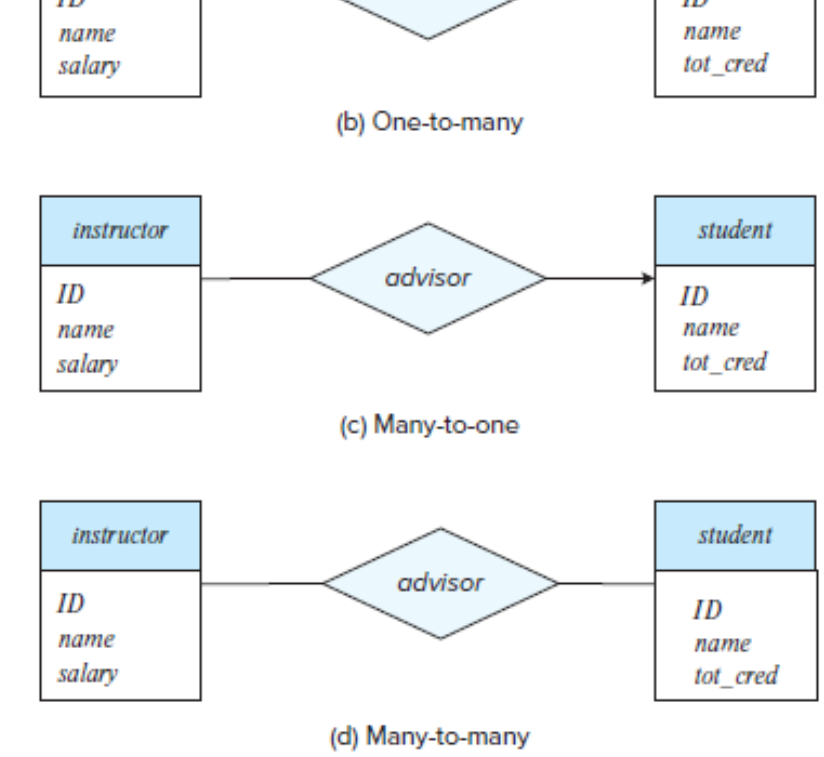
Entity-Relationship Diagram (ERD)

Mapping cardinality

They express **different** semantics

Different possible notations





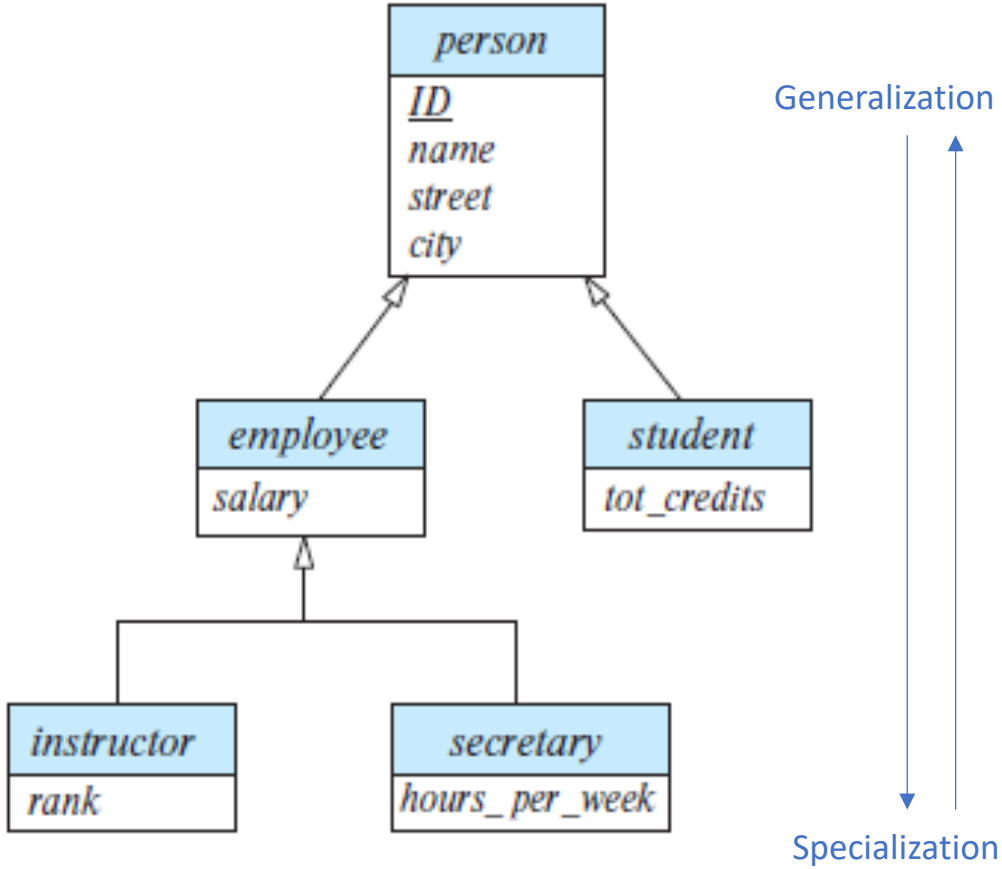
Entity-Relationship Diagram (ERD)

Specialization/Generalization

The process of designating sub-groupings within an entity set is called **specialization**.

For instance, the specialization of *person* allows us to distinguish among person entities according to whether they correspond to employees or students: in general, a person could be an employee, a student, both, or neither.

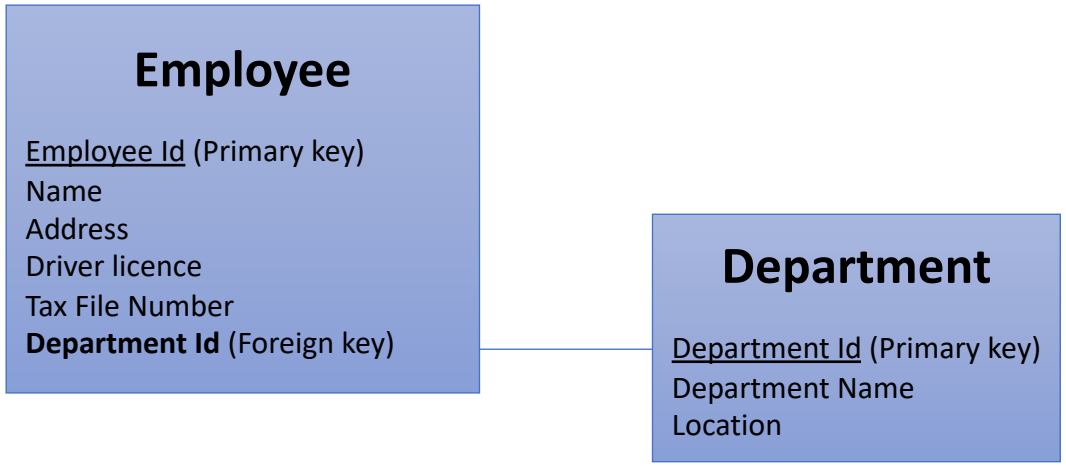
Generalization is the opposite process.



Entity-Relationship Diagram (ERD)

Foreign/Secondary Key

- **Foreign/Secondary Key** is an attribute that is used to identify an instance of another Entity.
- When a Primary Key of one entity is used in another Table, it is referred to as a 'Foreign Key' or 'Secondary Key'



IMPORTANT: in general terms, secondary key **DOES NOT** substitute relationships as it presents constraints in terms of cardinality.

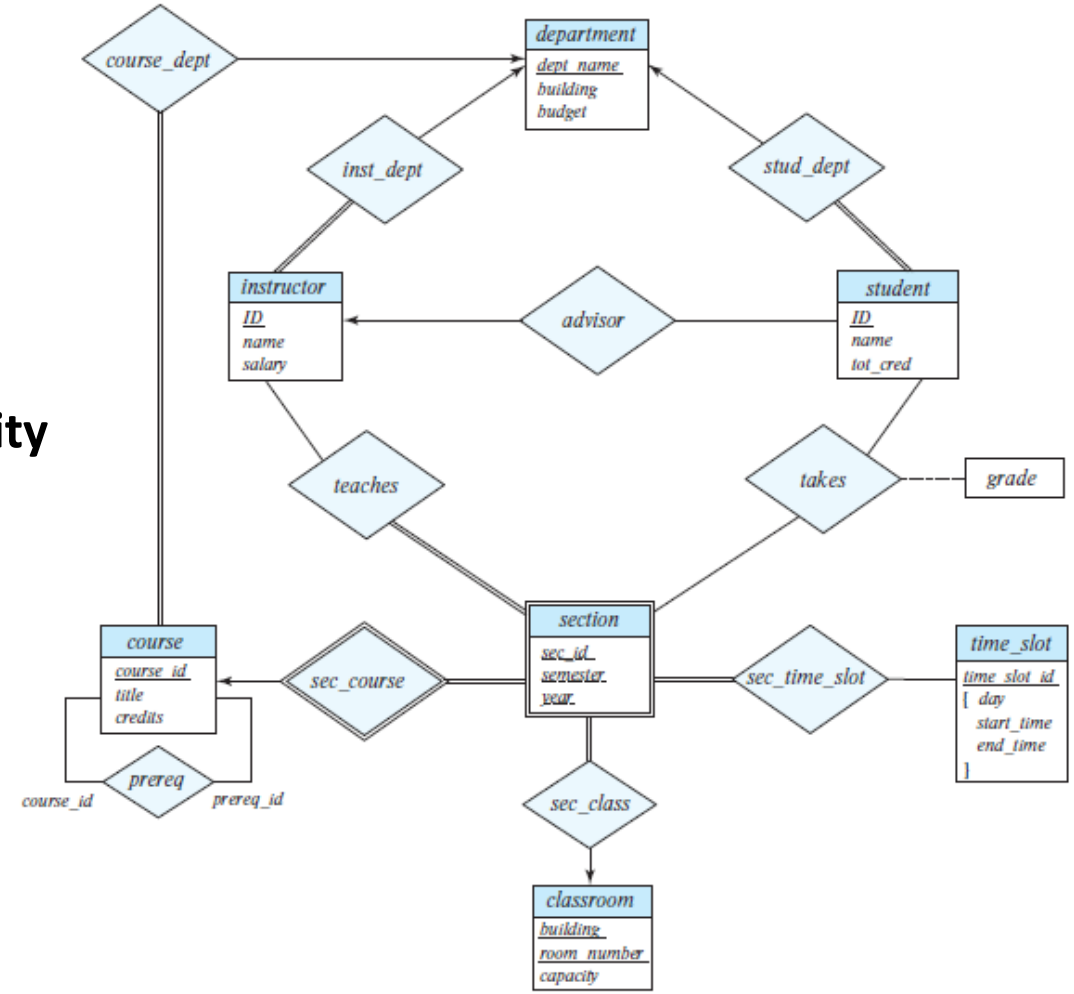
For instance, in the example, an employee can be associated to one department ONLY.

It allows direct associations between entities
Extensively used in DB optimization to minimize the number of tables

Entity-Relationship Diagram (ERD)

Dealing with complexity

- Structuring information is a **challenge**
- Even relatively simple systems present a certain **complexity**
- We need proper **skills & experience**

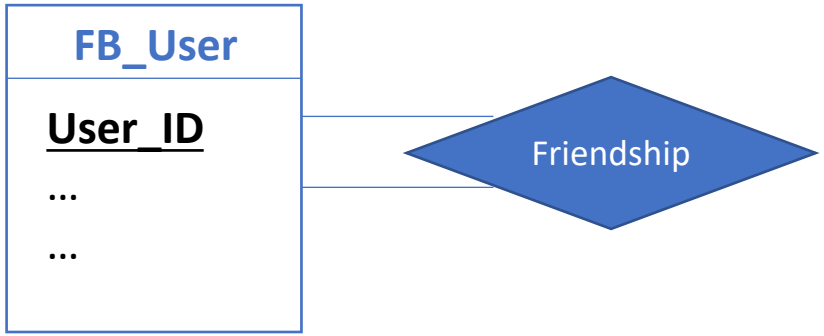


FAQs

FAQs

May I define a **recursive relationship**?

Of course!

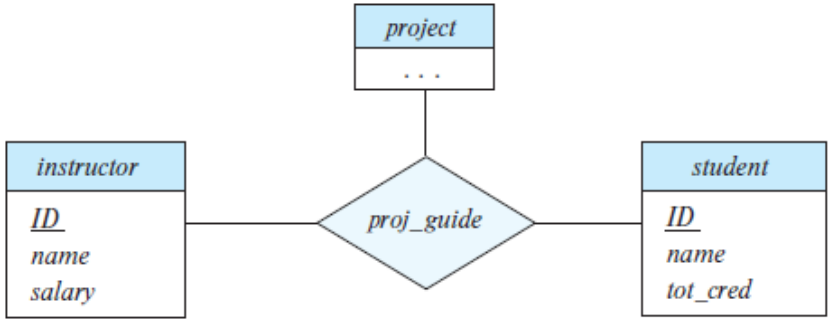


FAQs

May I define a **relationships involving more than 2 entities**?

Yes (by definition)

The number of entity sets that participate in a relationship set is the **degree of the relationship set**. A **binary relationship set** is of degree 2; a **ternary relationship set** is of degree 3. Most of the relationship sets in a database system are binary.




Thank You!