

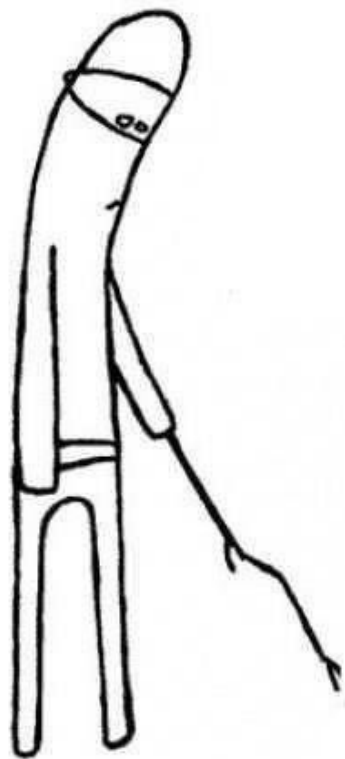
Databases - Tutorial 2

Conceptual Data Modeling

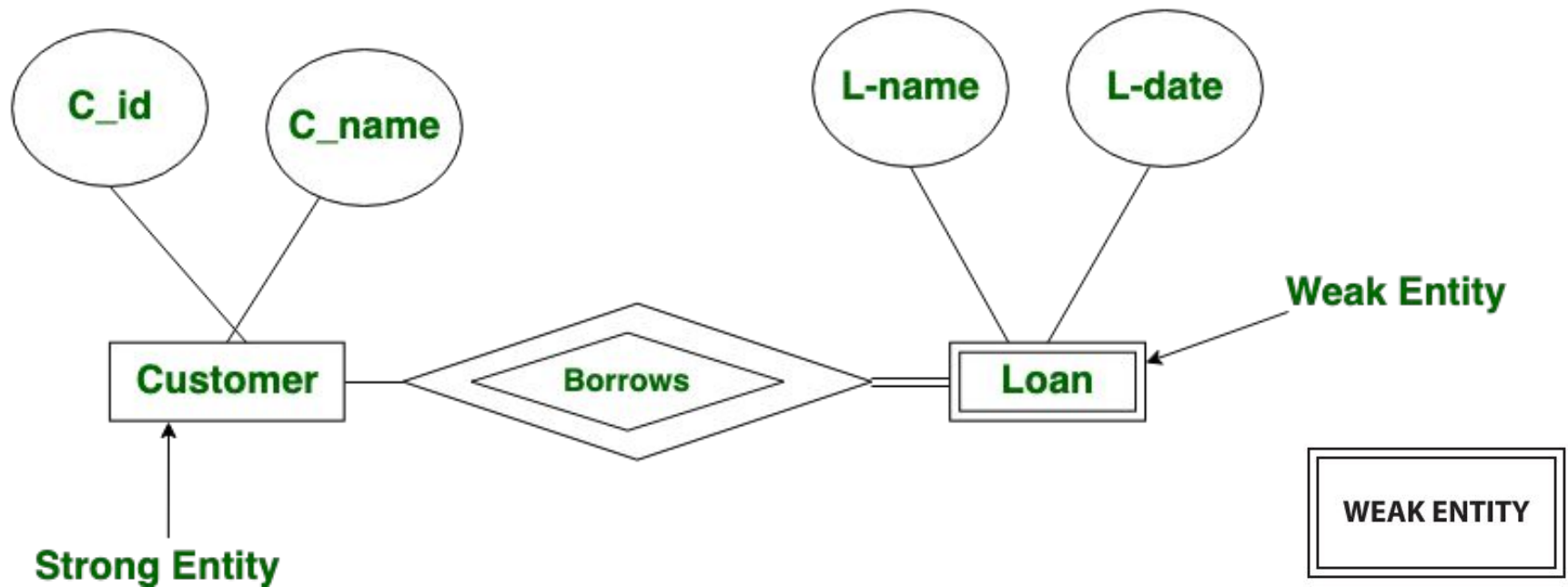
Hamza Salem - Innopolis University

Contents

- ERD review
- Unified Modeling Language (UML) | Class Diagrams



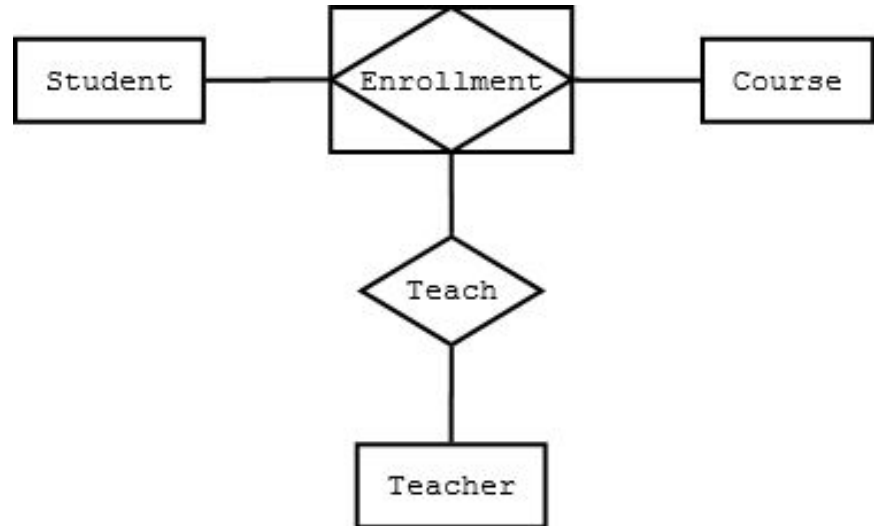
ENTITY

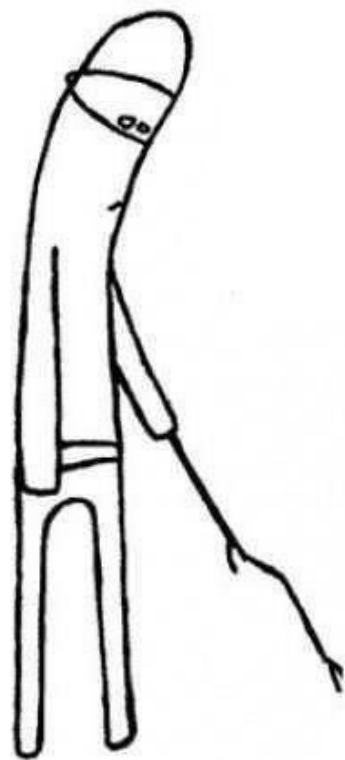


weak entity – an entity that cannot be uniquely identified by its attributes alone. The existence of a weak entity is dependent upon another entity called the owner entity. The weak entity's identifier is a combination of the identifier of the owner entity and the partial key of the weak entity.



associative entity – an entity used in a many-to-many relationship (represents an extra table). All relationships for the associative entity should be many





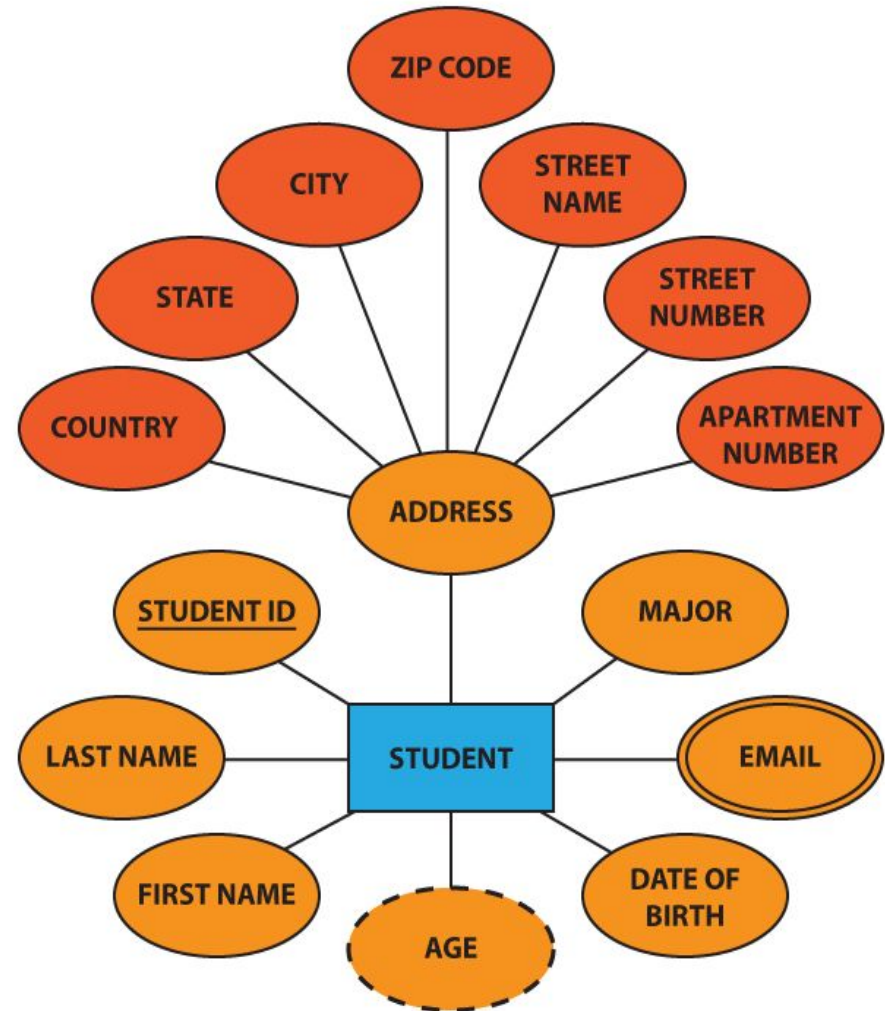
1- key attribute – an attribute that uniquely identifies a particular entity. The name of a key attribute is underscored.

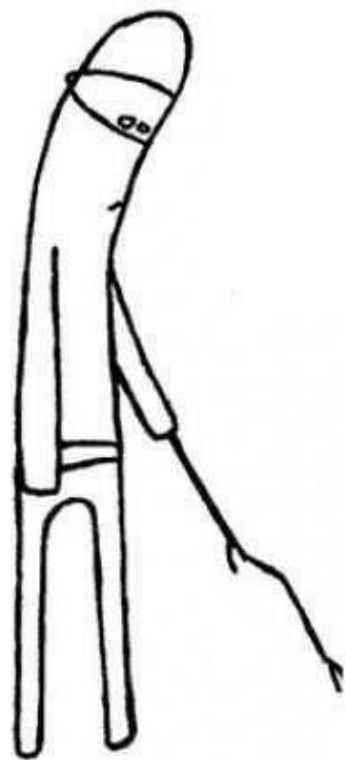
2- partial key attribute (discriminator) – an attribute that, when combined with the key attribute of the owner entity, provides a unique identification for the weak entity. We underline the discriminator with a dashed line.

3- multivalued attribute – an attribute that can have many values (there are many distinct values entered for it in the same column of the table). Multivalued attribute is depicted by a dual oval.

4- ?

5- ?





RELATIONSHIP

Relationships

Strong or weak :

- **strong relationship** – a relationship where entity is existence-independent of other entities, and PK of Child doesn't contain PK component of Parent Entity. A strong relationship is represented by a single rhombus.

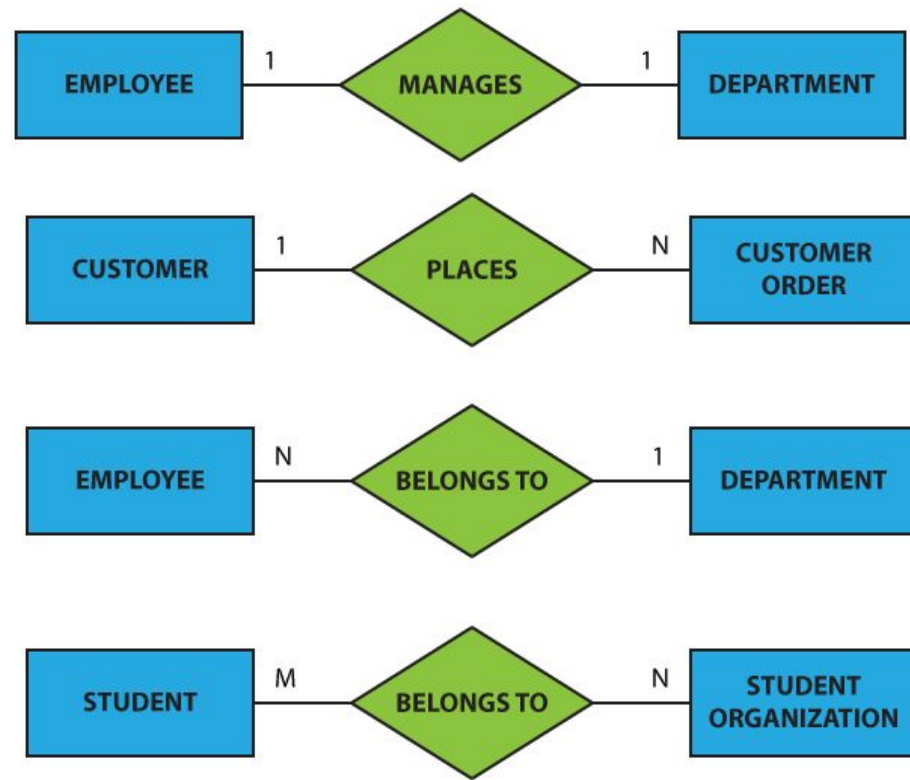
- **weak (identifying) relationship** – a relationship where Child entity is existence-dependent on parent, and PK of Child Entity contains PK component of Parent Entity. This relationship is represented by a double rhombus.

Optional or mandatory

- **mandatory** relationship is represented by a solid line.
- **optional** relationship is represented by a dashed line.

The **degree of relationship (cardinality)** is represented by characters "1", "N" or "M" usually placed at the ends of the relationships:

- **one-to-one (1:1)**
- **one-to-many (1:N)**
- **many-to-one (N:1)**
- **many-to-many (M:N)**

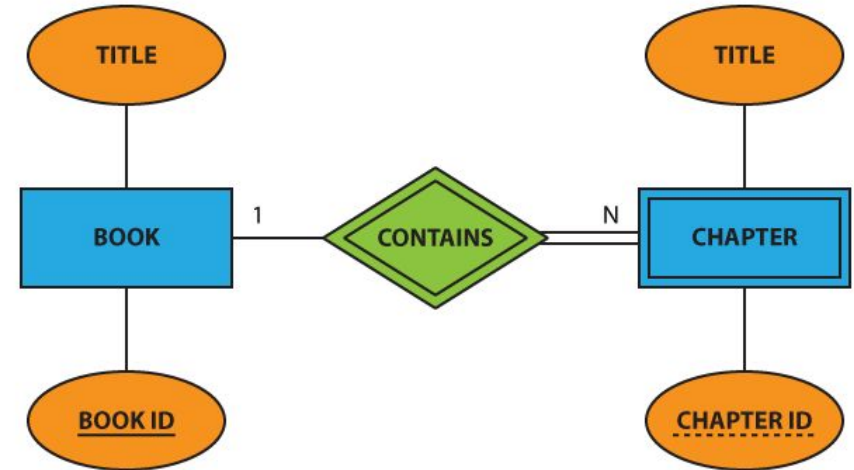
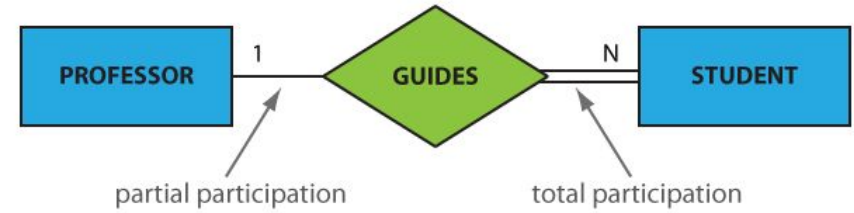


Relationships

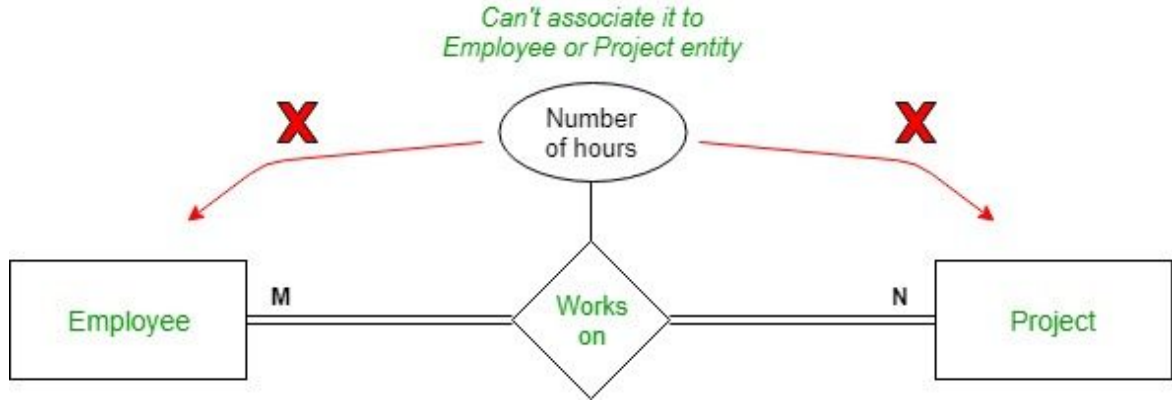
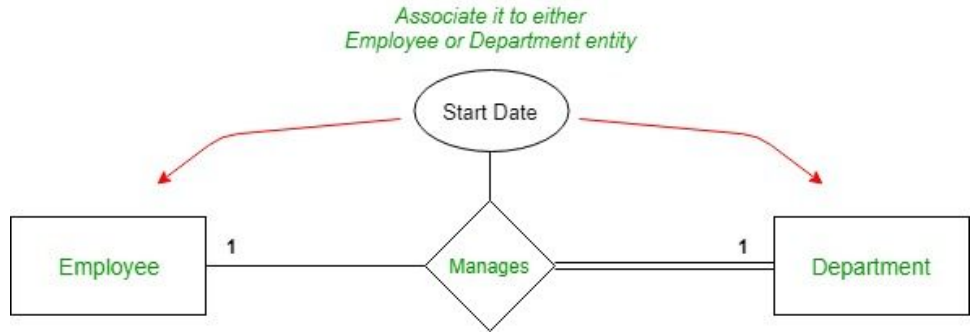
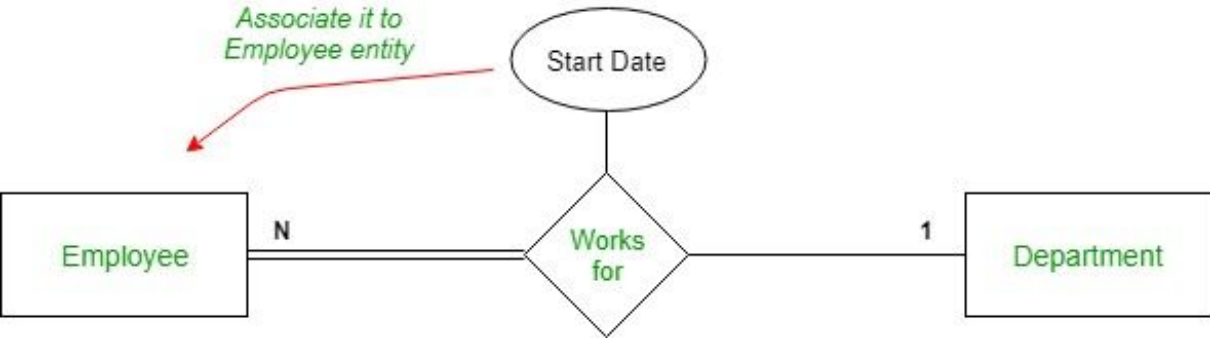
Participation constraints

An entity set may participate in a relation either totally or partially.

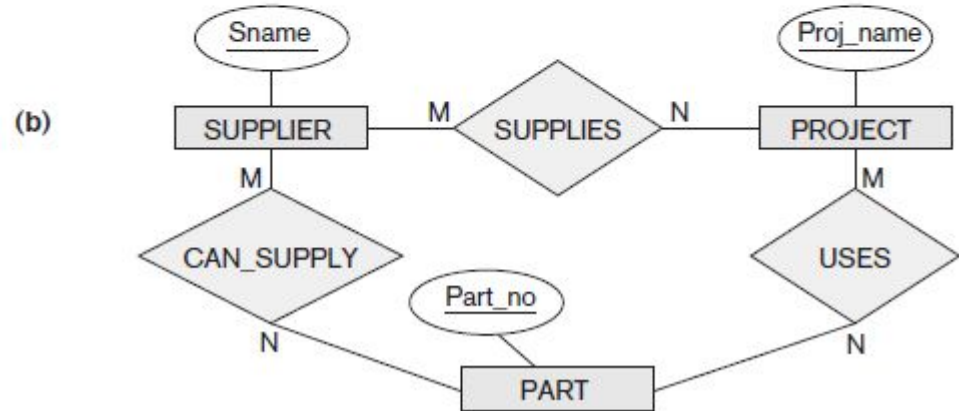
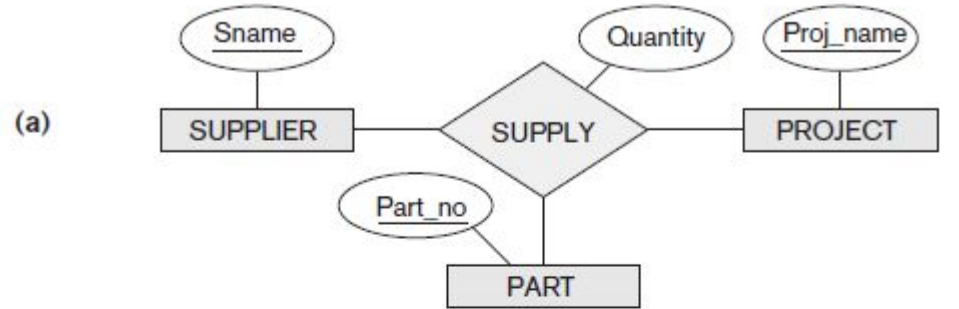
- **Total participation** means that every entity in the set is involved in the relationship, e.g., each student must be guided by a professor (there are no students who are not guided by any professor). In the Chen notation, this kind of relation is depicted as a double line.
- **Partial participation** means that not all entities in the set are involved in the relationship, e.g., not every professor guides a student (there are professors who don't). In the Chen notation, a partial participation is represented by a single line.



Attributes to Relationships



Choosing between Binary and Ternary (or Higher Degree) Relationships



Proper Naming of Schema Constructs

Singular names for entity types

Entity type and relationship type names are in uppercase letters

Attribute names have their initial letter capitalized

Role names are in lowercase letters

Nouns tend to give rise to entity type names, while verbs tend to indicate names of relationship types

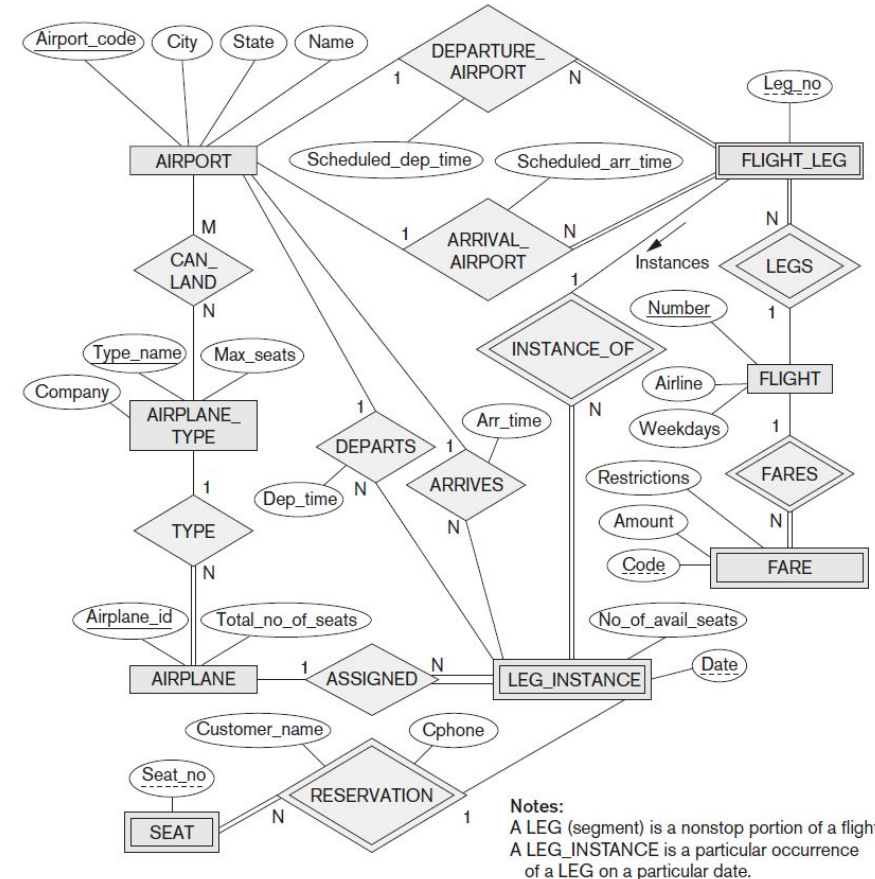
Attribute names generally arise from additional nouns

Make the ER diagram of the schema readable from left to right and from top to bottom, which is especially useful for representing relationships

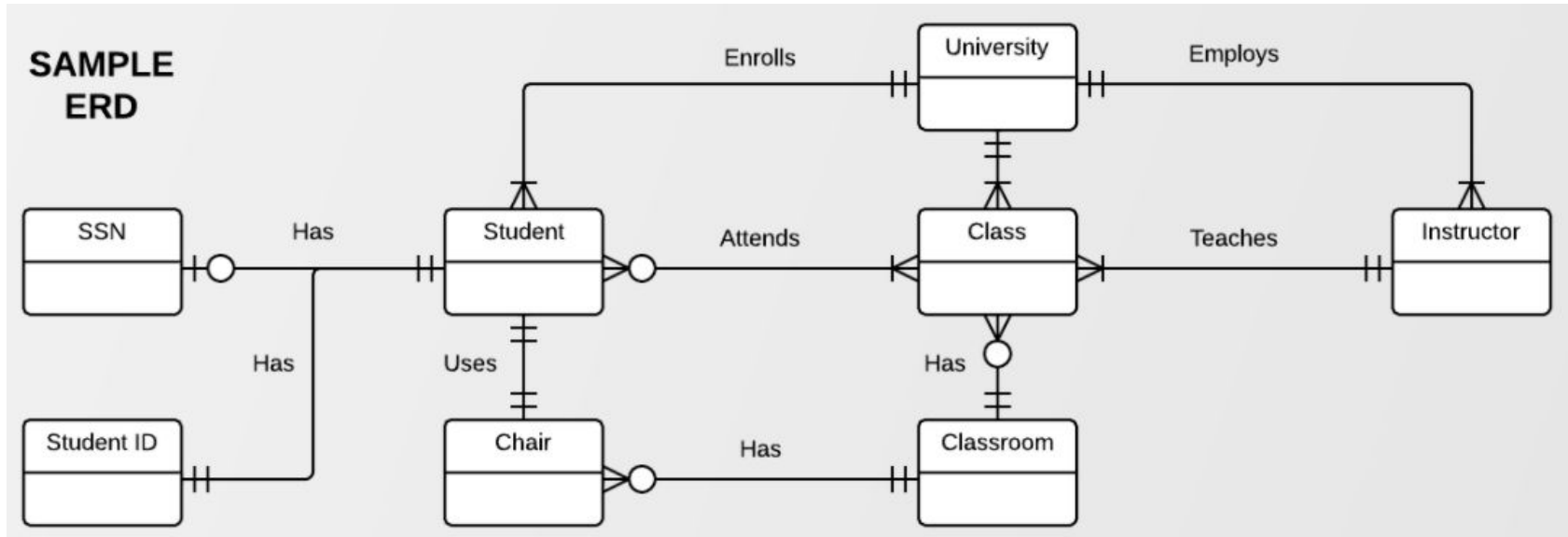


Doing reverse engineering






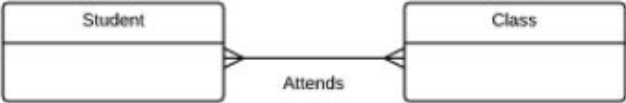







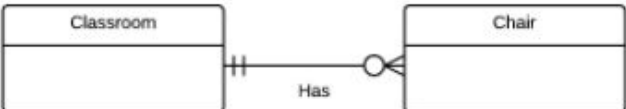
- Consider the ER diagram below, which shows a simplified schema for an AIRLINE reservations system.
- Extract from the ER diagram the requirements and constraints that produced this schema.
- Try to be as precise as possible in your requirements and constraints specification.



UML Notation



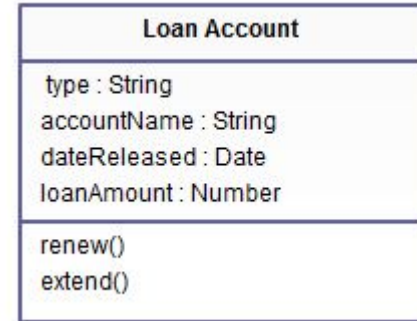
UML Notation

| Notation | Meaning | Example |
|--|------------------|--|
|  | Relationship |  |
|  | One |  |
|  | Many |  |
|  | One and ONLY One |  |
|  | Zero or One |  |
|  | One or Many |  |
|  | Zero or Many |  |

UML Notation

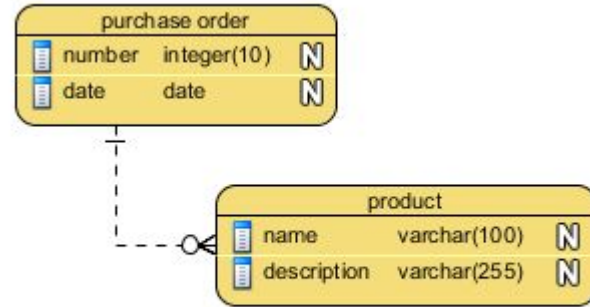
In the example, a class called “loan account” is depicted. Classes in **class diagrams** are represented by boxes that are partitioned into three:

1. The top partition contains the name of the class.
2. The middle part contains the class’s attributes.
3. The bottom partition shows the possible operations that are associated with the class.



Logical ERD

Logical ERD refines conceptual design to include/exclude primary entities and relate them.



Physical ERD

The most detailed design that is readily adaptable to physical database

