# ER and EER Diagram

Trishla Shah

trishla.shah@smu.ca

# Why should I learn ER Diagram

# ER Diagram

- An entity-relationship diagram (ERD) is a data modeling technique that graphically illustrates an information system's entities and the relationships between those entities.

- An entity-relationship diagram (ERD) is crucial to creating a good database design.

- It is used as a high-level logical data model, which is useful in developing a conceptual design for databases.

# Main Elements of ERD
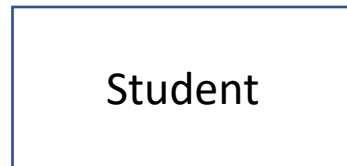
- Entities
- Attributes
- Relationships

# Entities

- An entity is a real-world item or concept that exists on its own.

- Entities are equivalent to database tables in a relational database, with each row of the table representing an instance of that entity.

  symbol (Notation): Entities are represented by rectangles.

Entity

# Example

## ERD

Student

## Relational database

| STUDENT | |
|---|---|
| StuNo | StuName |
| 1 | Alice |
| 2 | Bob |
| 3 | Chris |
| 4 | David |
| 5 | Eve |
| 6 | Fred |

# Types Entities

- Entity

- Weak Entity: A weak entity is an entity that depends on the existence of another entity. In more technical terms it can be defined as an entity that cannot be identified by its own attributes.

Example: Students' grade will be meaningless without students record so it depends on the existence of the student table.

```
┌──────────────┐        ┌──────────────────┐
│              │        │ ┌──────────────┐ │
│   Student    │────────│ │Student Grades│ │
│              │        │ └──────────────┘ │
└──────────────┘        └──────────────────┘
```

# Question
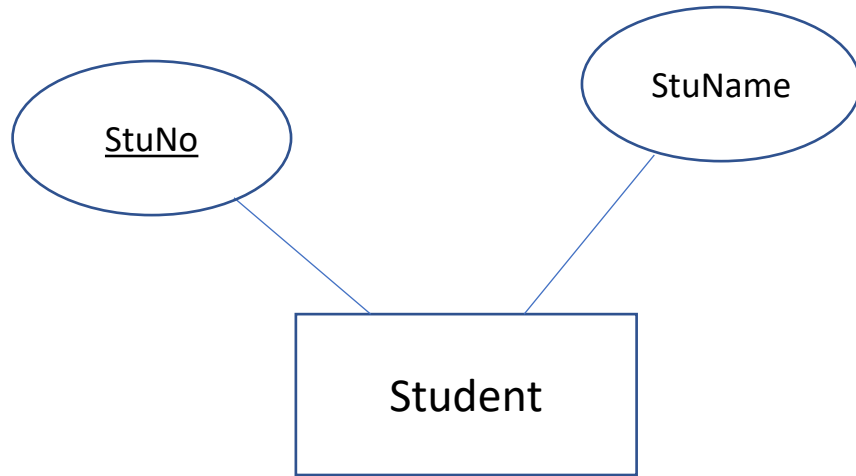
- Give two example of weak entity.

# Attributes

- An attribute of an entity is a particular <u>property</u> that describes the entity.

- symbol (Notation): Attributes are represented by ovals.

# Example

## ERD



StuNo

StuName

Student

## Relational database

| STUDENT | |
|---|---|
| StuNo | StuName |
| 1 | Alice |
| 2 | Bob |
| 3 | Chris |
| 4 | David |
| 5 | Eve |
| 6 | Fred |

# Types of Attributes

- Key Attribute
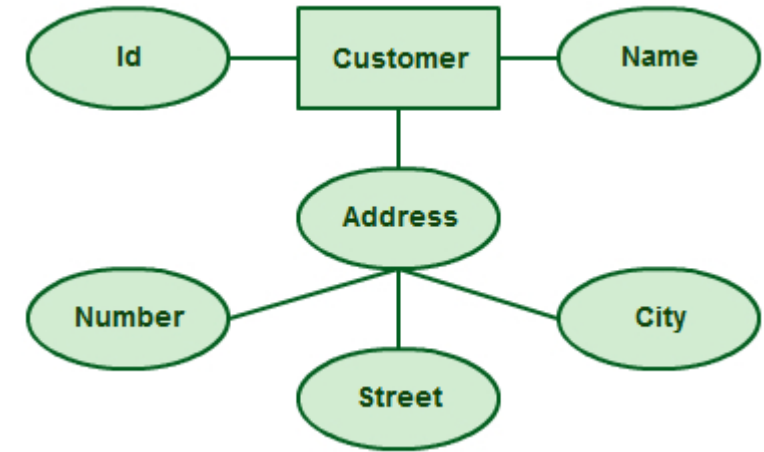- Multivalued attribute
- Derived Attribute

# Key Attribute

- A key attribute is the unique, distinguishing characteristic of the entity.

- For example, an employee's social security number might be the employee's key attribute.
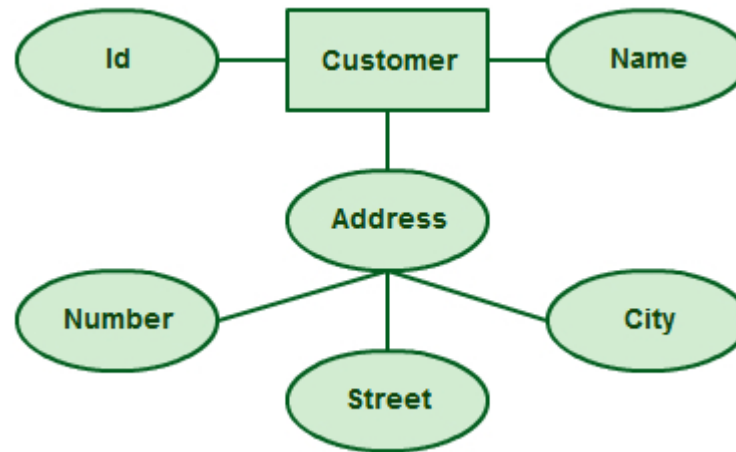
- Symbol (Notation):    

# One More Example



- Here customer entity can have the attributes id, name address.

- But the attribute "customer address" can also have the attributes number, street, city, and state. These are called composite attributes.

- Note: Some top level ER diagrams do not show attributes for the sake of simplicity

# Question

- Generate Relational Database Schema based on the following diagram.

# Multivalued Attribute

- A multivalued attribute can have more than one value.

- For example, an employee entity can have multiple skill values.

- Symbol (Notation):      Attribute

# One More Example

- A teacher entity can have multiple subject values.

# Question

- Generate Relational Database Schema based on the following diagram. You can add more attributes such as id, teacher name, subjects, etc.
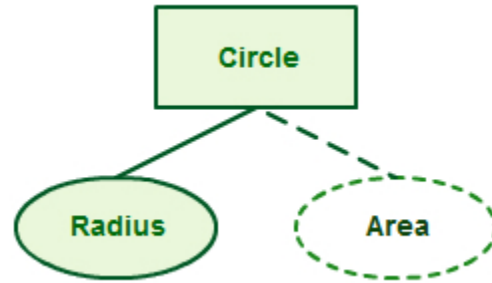
# Derived Attribute

- A derived attribute is based on another attribute.

- For example, an employee's monthly salary is based on the employee's annual salary.

- Symbol (Notation):


Attribute

# One More Example

- The area can be derived from the radius.

# Question:

- Give two example of multivalued attribute.
- Give two example of derived attribute.

# Relationships

- A relationship describes how entities interact.

- Relationships are represented by diamond shapes and are labeled using verbs (It is also known as Actions).

- Symbol (Notation):   Relationship

# Example

- For example, the entity "Carpenter" may be related to the entity "table" by the relationship "builds" or "makes".

# Question

- If the same entity participates more than once in a relationship then how to show it? For example, an employee can be supervisor and be supervised (Hint: Self-join).

# Answer

- If the same entity participates more than once in a relationship then how to show it? For example, an employee can be supervisor and be supervised (Hint: Self-join).

- It is known as a recursive relationship.
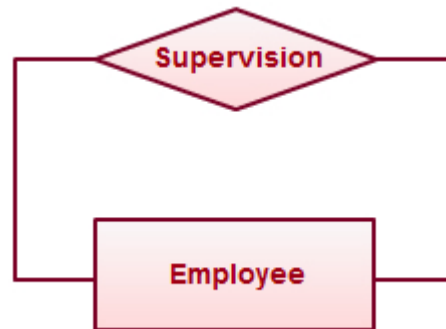
# Ordinality and Cardinality

- These further define relationships between entities by placing the relationship in the context of numbers.

- The relationship, in this case, follows a "one to many" model.

# Types

- 1:N (n=0,1,2,3,..)  → One to Zero or more

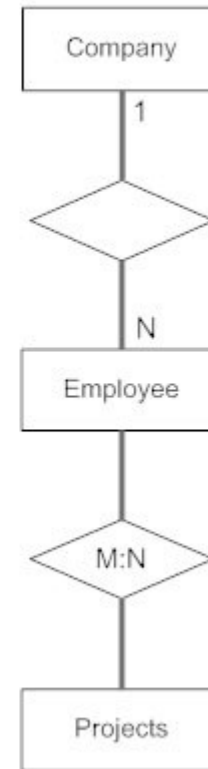- M:N (m and n=0,1,2,3,…) → Zero or more to Zero or more (many to many)

- 1:1 → One to one

# Notation/Symbol

Ordinality - describes the minimum ──► M:N ◄── Cardinality - describes the maximum

1:N (n=0,1,2,3...)
one to zero or more

M:N (m and n=0,1,2,3...)
zero or more to zero or more
(many to many)

1:1
one to one

# One More Example

- For example, In an email system, for example, one account can have multiple contacts.

# Question

- Give Example one of 1:1, 1:N, and M:N

# Steps to create ER Diagrams

- Step 1: Identify all the entities
- Step 2: Identify relationships
- Step 3: Add attributes

# Step 1: Identify all the entities

- Identify all the entities in the system.

- An entity should appear only once in a particular diagram.

-  Create rectangles for all entities and name them properly.

# Step 2: Identify relationships

- Identify relationships between entities.
- Connect them using a line and add a diamond in the middle describing the relationship.

# Step 3: Add attributes

- Add attributes for entities.
- Give meaningful attribute names so they can be understood easily.

# Example - Question

- University wants to store students information (StuName). They also want to store the courses taken by each student along with the supervisors' name and lecturers' name for each course.
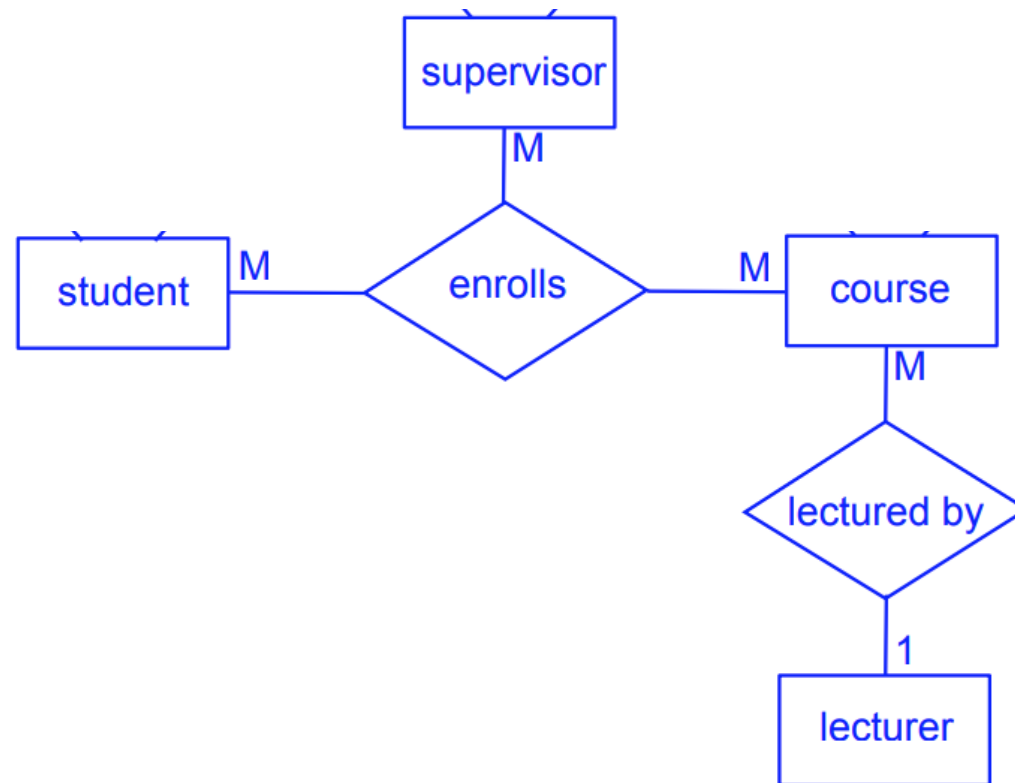
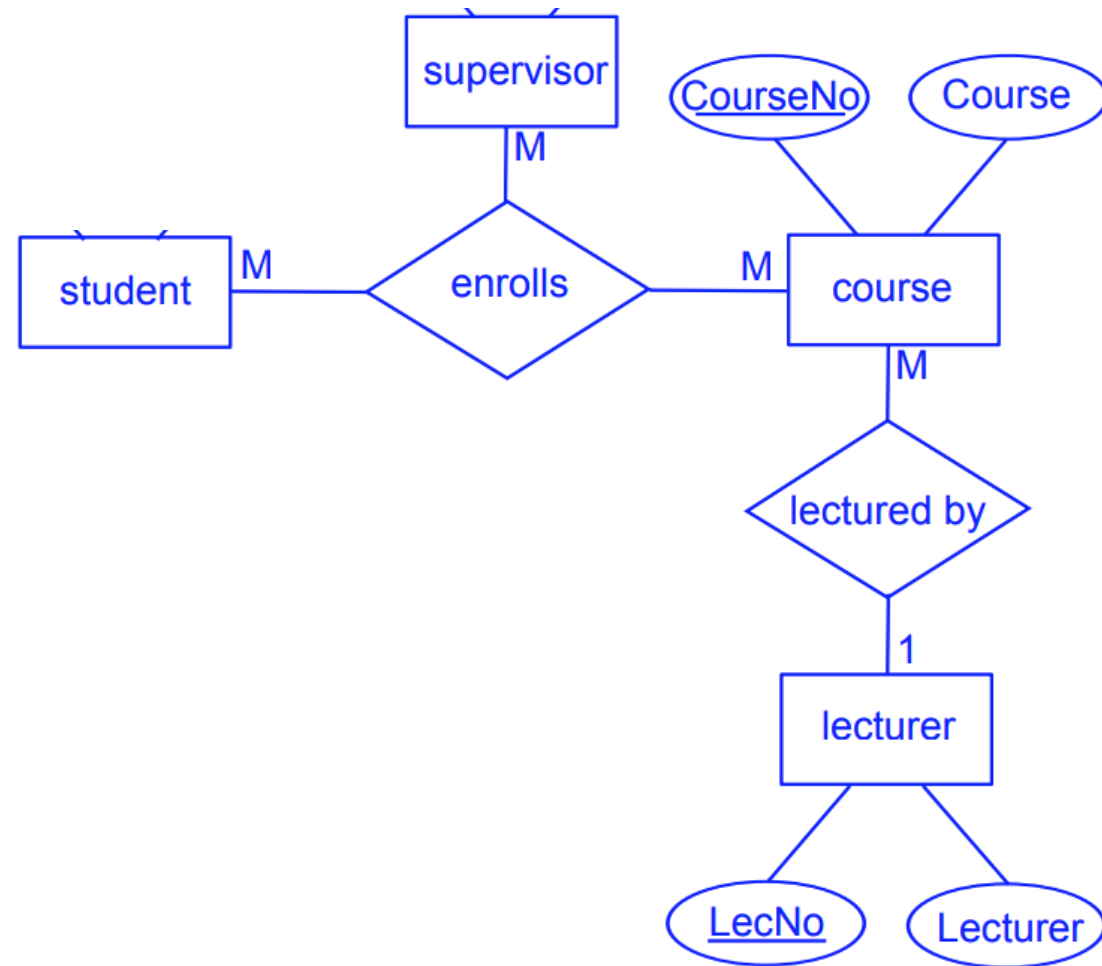# Step 1: Identify all the entities

supervisor

student

course

lecturer

# Step 2: Identify relationships

# Step 3: Add attributes

# Relate With Relational Database

**STUDENT-TEACHING**

| StuNo | Name | Course1 | Lecturer1 | Supervis1 | Course2 | Lecturer2 | Supervis2 | Course3 | Lecturer3 | Supervis3 |
|---|---|---|---|---|---|---|---|---|---|---|
| 1 | Alice | maths | Gauss | Black | structures | Brunel | Brown | mechanics | Brunel | Brown |
| 2 | Bob | maths | Gauss | White | mechanics | Brunel | Brown | | | |
| 3 | Chris | mechanics | Brunel | Greene | structures | Brunel | Brown | inf. eng. | Shannon | White |
| 4 | David | inf. eng. | Shannon | Black | mechanics | Brunel | Greene | | | |
| 5 | Eve | maths | Gauss | White | inf. eng | Shannon | White | structures | Brunel | Greene |
| 6 | Fred | inf. eng. | Shannon | Black | | | | | | |

**SUPERVISOR**

| SupNo | SupName |
|---|---|
| 1 | Black |
| 2 | White |
| 3 | Brown |
| 4 | Greene |

**STUDENT**

| StuNo | StuName |
|---|---|
| 1 | Alice |
| 2 | Bob |
| 3 | Chris |
| 4 | David |
| 5 | Eve |
| 6 | Fred |
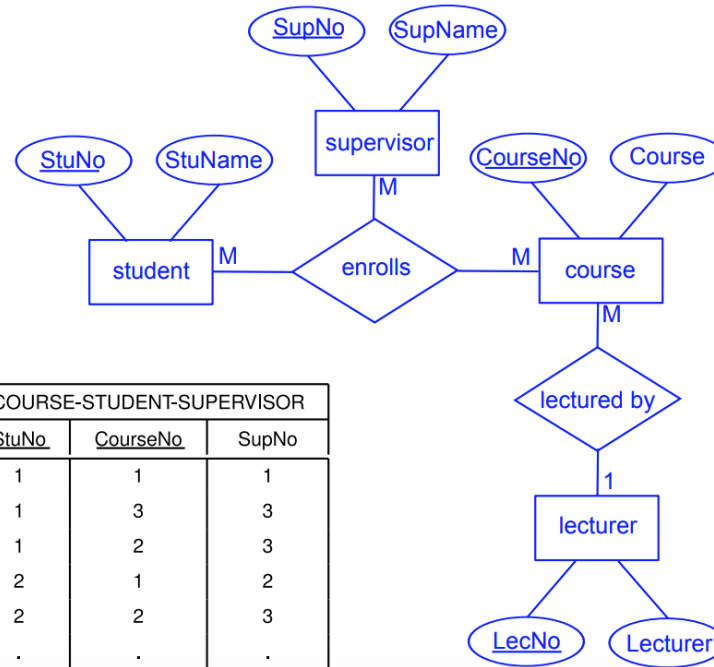
**COURSE-STUDENT-SUPERVISOR**

| StuNo | CourseNo | SupNo |
|---|---|---|
| 1 | 1 | 1 |
| 1 | 3 | 3 |
| 1 | 2 | 3 |
| 2 | 1 | 2 |
| 2 | 2 | 3 |
| . | . | . |
| . | . | . |
| . | . | . |

**COURSE**

| CourseNo | Course |
|---|---|
| 1 | maths |
| 2 | mechanics |
| 3 | structures |
| 4 | inf. eng. |

**COURSE-LECTURER**

| CourseNo | LecturerNo |
|---|---|
| 1 | 1 |
| 2 | 2 |
| 3 | 2 |
| 4 | 3 |

**LECTURER**

| LecturerNo | Lecturer |
|---|---|
| 1 | Gauss |
| 2 | Brunel |
| 3 | Shannon |

ER diagram:

SupNo, SupName — supervisor

StuNo, StuName — student

CourseNo, Course — course

student M — enrolls — M course (supervisor M)

course M — lectured by — 1 lecturer

LecNo, Lecturer

# Question

- Draw ER Diagram (you can take any example):
  - ➤ It should have at least four entities
  - ➤ It should have at least two derived attributes
  - ➤ It should have at least one multi valued attribute
  - ➤ It should have at least one 1:1 relationship
  - ➤ It should have at least one 1:N or M:N relationship