A primary goal of database design is to decide what tables to create. Usually, there are two principles:

1. Capture all the information that needs to be captured by the underlying application.
2. Achieve the above with little redundancy.

The first principle is enforced with an entity relationship (ER) diagram, while the second with normalization.

This lecture focuses on the ER diagram.
An ER diagram is a pictorial representation of the information that can be captured by a database. Such a “picture” serves two purposes:

- It allows database professionals to describe an overall design concisely yet accurately.
- (Most of) it can be easily transformed into the relational schema.
An entity is an atomic object that needs to be represented in the database. An entity set is a set of entities with common attributes.

For example, a professor is an entity. PROF is an entity set with all the professors.

The following shows how to describe in an ER-diagram the entity set PROF with attributes pid, hkid, dept, rank, salary, with pid being the primary key.
A **relationship** is an association among several entities that needs to be represented in the database. We will denote the relationship as \((e_1, e_2, ..., e_n)\), where \(e_1, ..., e_n\) are the entities participating in the relationship.

A **relationship set** is a set \(R\) of relationships \((e_1, ..., e_n)\), where each \(e_i\) comes from the same entity set \(E_i\), for \(1 \leq i \leq n\).

**Example:** The following shows a relationship set \(R\) between entity sets \(A\) and \(B\).

\[
R = \{(a_1, b_1), (a_1, b_4), (a_2, b_2), (a_2, b_4), (a_3, b_3)\}.
\]
The next few slides will discuss binary relationship sets, i.e., $n = 2$. For such relationship sets, we can impose two types of constraints.

**Cardinality constraint:**
- One-to-one.
- One-to-many (or conversely, many-to-one).
- many-to-many.

**Participation constraint:**
- Total.
- Partial.

We will first talk about cardinality constraints.
A relationship set $R$ between entity sets $A$ and $B$ is one-to-one if every entity in $A$ and $B$ can participate in at most one relationship in $R$.

Example: Husbands and wifes.
A relationship set $R$ between entity sets $A$ and $B$ is **one-to-many** if every entity in $A$ can participate in any number of relationships in $R$, but an entity in $B$ can participate in at most one relationship in $R$.

**Example:** Parents and Children.

**Many-to-one** is defined analogously.
A relationship set $R$ between entity sets $A$ and $B$ is many-to-many if every entity in $A$ and $B$ can participate in any number of relationships in $R$.

Example: Students and classes.
Cardinality constraint:
- One-to-one.
- One-to-many (or conversely, many-to-one).
- many-to-many.

Participation constraint:
- Total.
- Partial.

Next we talk about participation constraints.
Let $R$ be a relationship set between entity sets $A$ and $B$. The participation of $A$ is **total** if every entity of $A$ must participate in at least one relationship in $R$. Otherwise, the participation of $A$ is **partial**. Likewise, we can define total or participation of $B$.

**Example:** In the following relationship, the participation of $A$ is total, while that of $B$ is partial.

![Entity Relationship Diagram](image)

Patents and professors.
Basic representation of a binary relationship set $R$ between entity sets $A$ and $B$.

The line connecting $A$ and $R$ may:

- Carry an arrow: Meaning one (as in one-to-many).
- Not carry an arrow: Meaning many.
- Be a double line: Meaning total participation.
- Be a single line: Meaning partial participation.
Examples:

Many-to-many, partial participation on both

One-to-one, total (partial) participation on left (right)

Many-to-one, total (partial) participation on left (right)
Representation a 3-way relationship set:

Example: Professors, students, and projects.
We can associate a relationship set with attributes:

![Entity Relationship Diagram](image)

What is its difference from the design below?

![Entity Relationship Diagram](image)
Revisit the ER diagram on Slide 4 to understand all of its components.