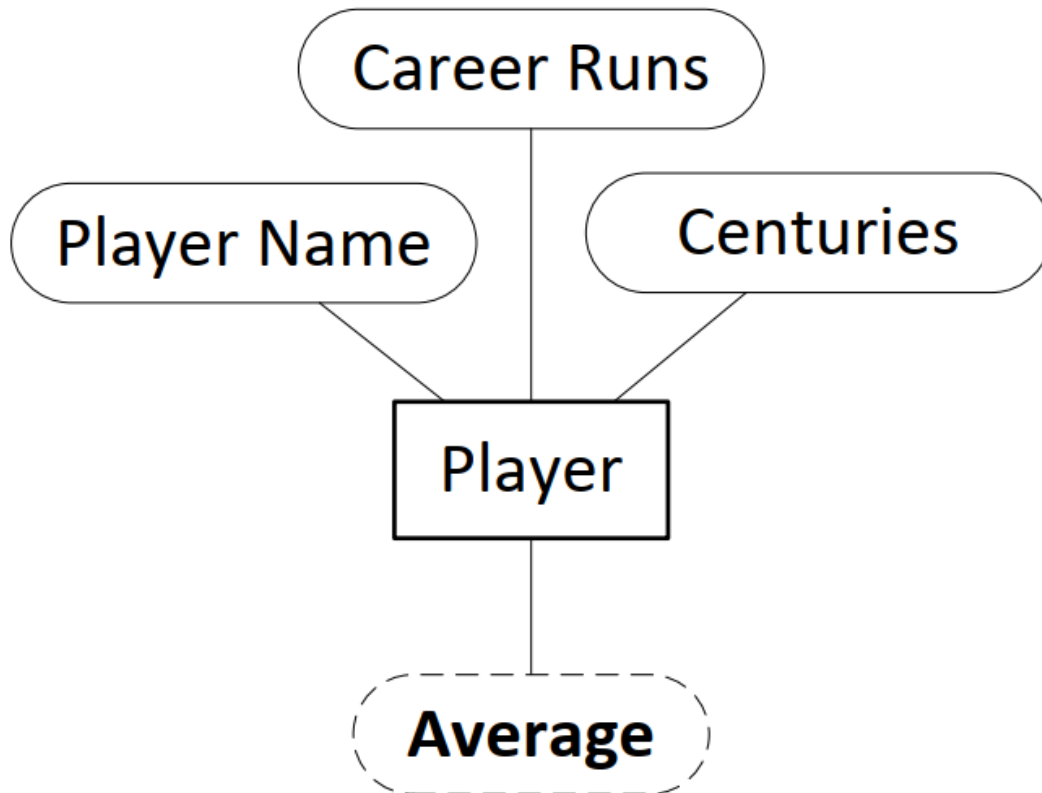


Derived Attributes

- Value can be computed from other attributes
 - ▣ Example: Age, given Date of Birth

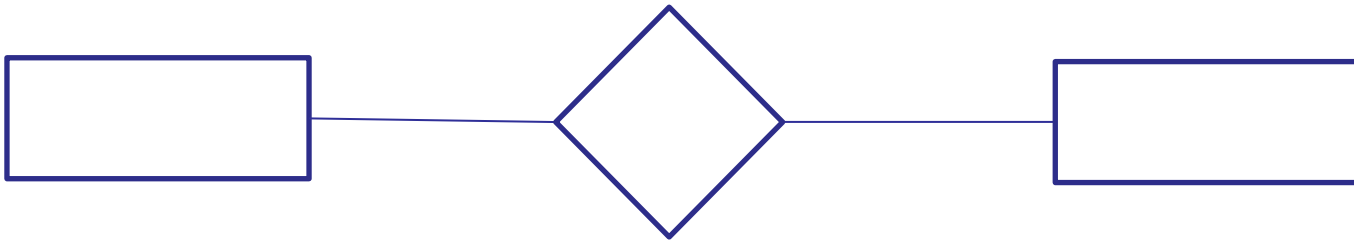


Player
Player Name
Career Runs
Centuries
[Average]

Basic Components of ERD

Relationship:

- Relationships are represented by diamond shaped box.
- Name of the relationship is written in the diamond-box.
- All entities (rectangles), participating in relationship, are connected to it by a line.

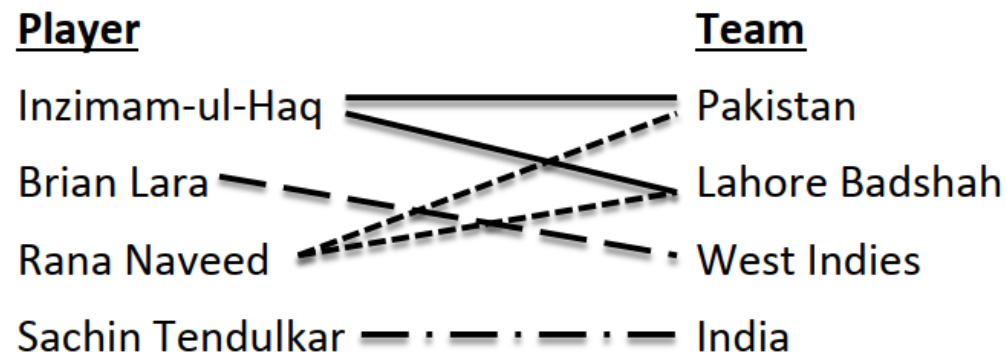


Relationship

- Relationship Type: Association between entity types
 - Modeled as a connecting labeled line (or diamond) between entity types.



- Relationship Instance: Association between entity instances
 - Also modeled as connecting lines.



Cardinality of Relationships

Types of Relationships:

Three types of relationships can exist between entities:

- One-to-one relationship (1:1):

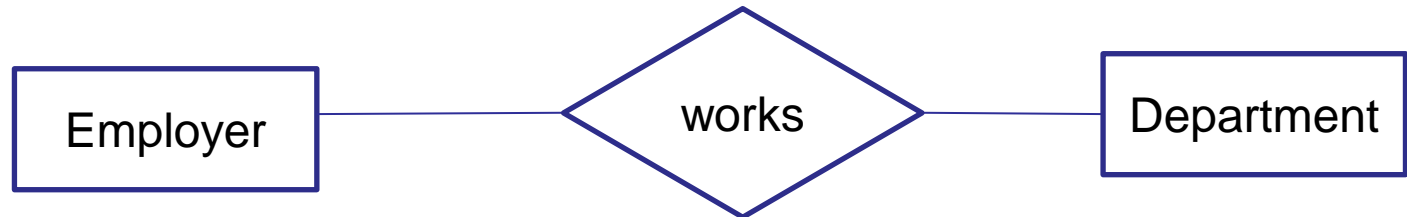
One instance in an entity (parent) refers to one and only one instance in the related entity (child)

- One-to-many relationship (1:M)

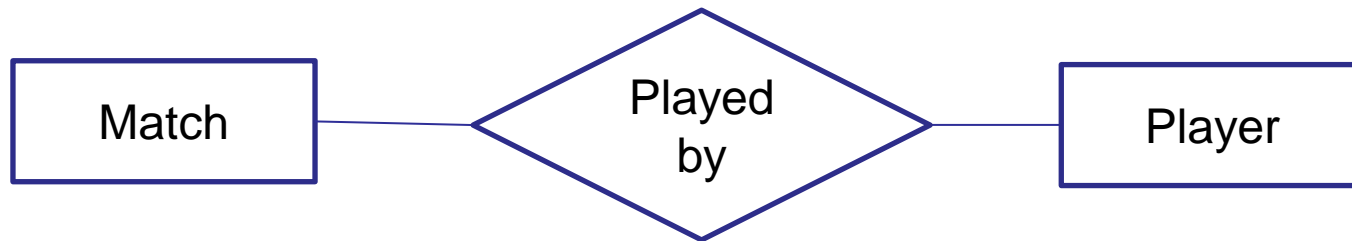
One instance in an entity (parent) refers to one or more instances in the related entity (child)

Cardinality of Relationships

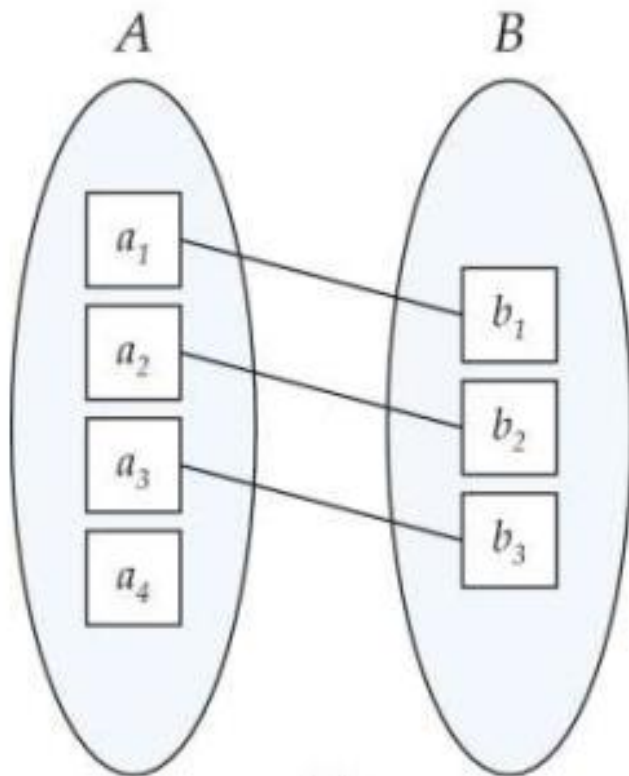
- One-to-one relationship (1:1):



- One-to-many relationship(1:M)

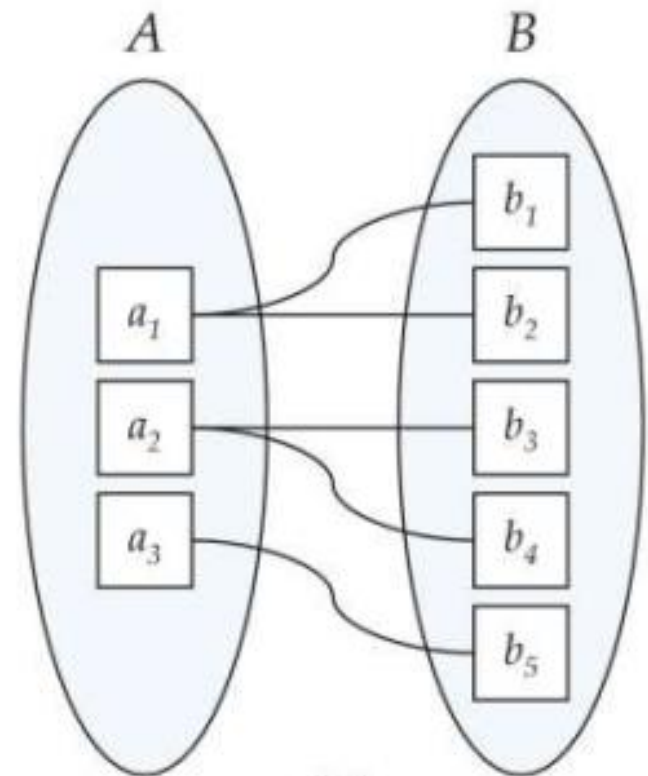


Cardinality of Relationships



(a)

One to one



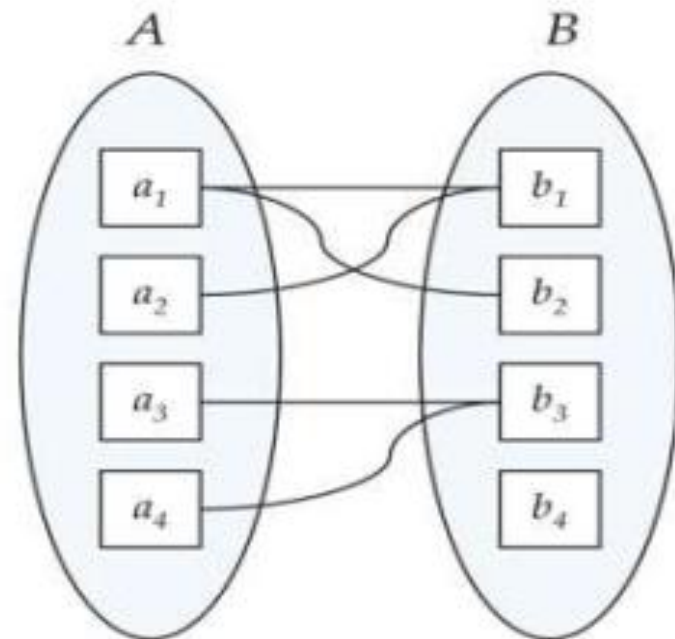
(b)

One to many

Cardinality of Relationships

Many-to-many relationship (M:N)

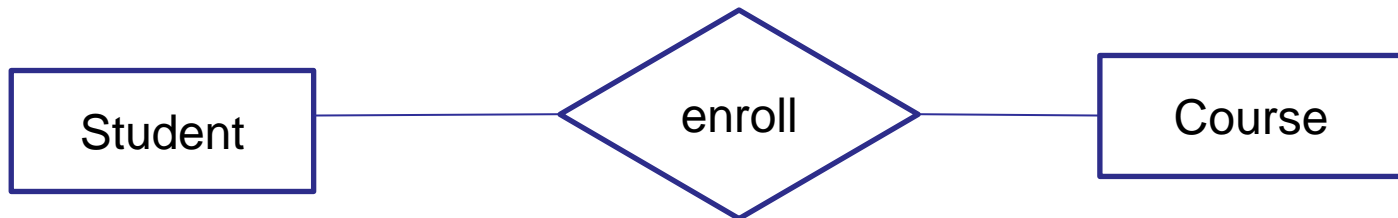
- This relationship exists when one instance of the first entity (parent) can relate to many instances of the second entity (child), and one instance of the second entity can relate to many instances of the first entity.



Many to many

Cardinality of Relationships

Many-to-many relationship (M:N)



Cardinality of Relationships

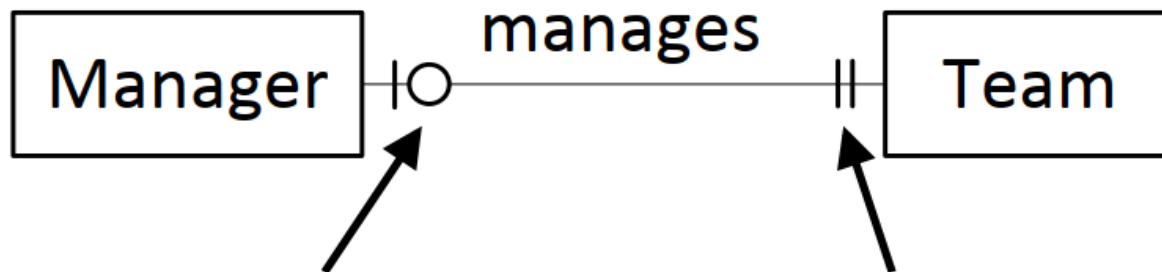
One to One – A single row of first table associates with single row of second table. For example, a relationship between person and passport table is one to one because a person can have only one passport and a passport can be assigned to only one person.

One to Many – A single row of first table associates with more than one rows of second table. For example, relationship between customer and order table is one to many because a customer can place many orders but an order can be placed by a single customer alone.

Many to Many – Many rows of first table associate with many rows of second table. For example, relationship between student and course table is many to many because a student can take many courses at a time and a course can be assigned to many students.

One-to-One relationship

- One entity instance associated with only (maximum) one target entity instance



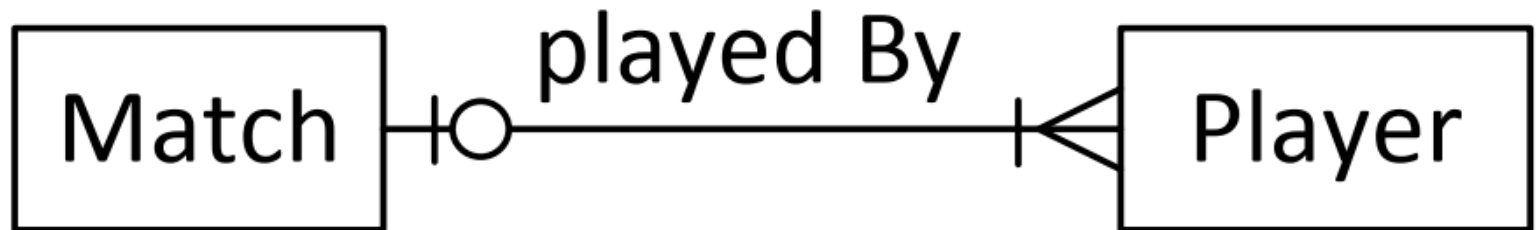
A manager is associated with only one team at most, or there is no manager.

One team is managed by only one manager at most or not managed at all.



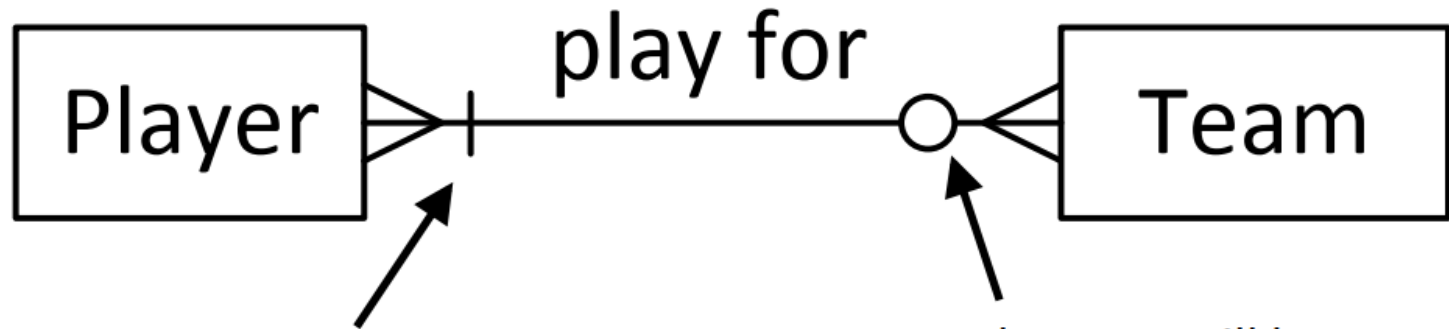
One-to-Many relationship

- One entity instance associated with optionally or mandatory many target entity instances.



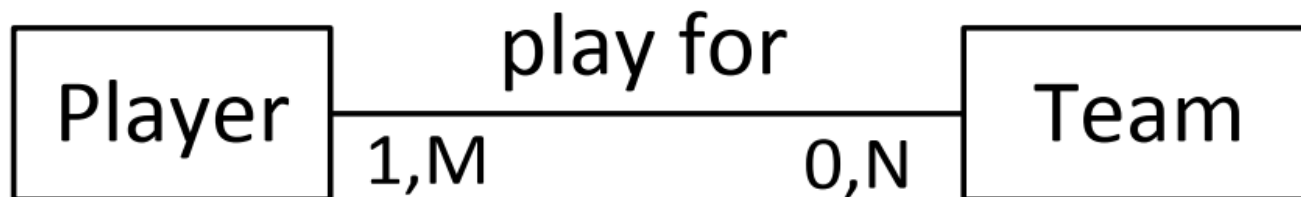
Many-to-Many relationship

- Many entity instances associated with either optionally or mandatory many target entity instances



One or more players are associated with none or more teams.

Each teams will have one or more players.



Strong vs Weak Entities

- **Strong Entity Type:** One that exists independently of other entity types.
 - ▣ Strong entity instances always have a **unique identifier**
 - ▣ Identifier is **underlined with single-line**
 - ▣ **Examples:** Student, Employee, Course
- **Weak Entity Type:** One whose existence depends on a strong entity (or Identifying Owner).
 - ▣ It only has a partial identifier
 - ▣ Partial identifier is **underlined with double-line**
 - ▣ Entity **box has double line**
 - ▣ **Example:** Student versus StudentAddress

Advantages and Usage

- **Advantages:**

1. ERD tells us that how many tables you need and what would be the relationship between them (you also have to do Normalization to know finally how many tables would be in your database but still first step is ERD).
2. ERD is simple and understandable representation of a database. It helps a lot to understand the whole database.

- **Usage:**

1. An ERD leads to ERM, means when ever you need to build a database with tables, firstly, you need to create an ERD.
2. Crow's foot notation is used most of all because its easy to understand in implementation point of view.

Different types of keys

Primary Key:

- A primary key is a candidate key that is most appropriate to be the main reference key for the table.
- As its name suggests, it is the primary key of reference for the table and is used though out the database to help establish relationships with other tables.
- **The primary key must contain unique values, must never be null and uniquely identify each record in the table.**

Primary Keys

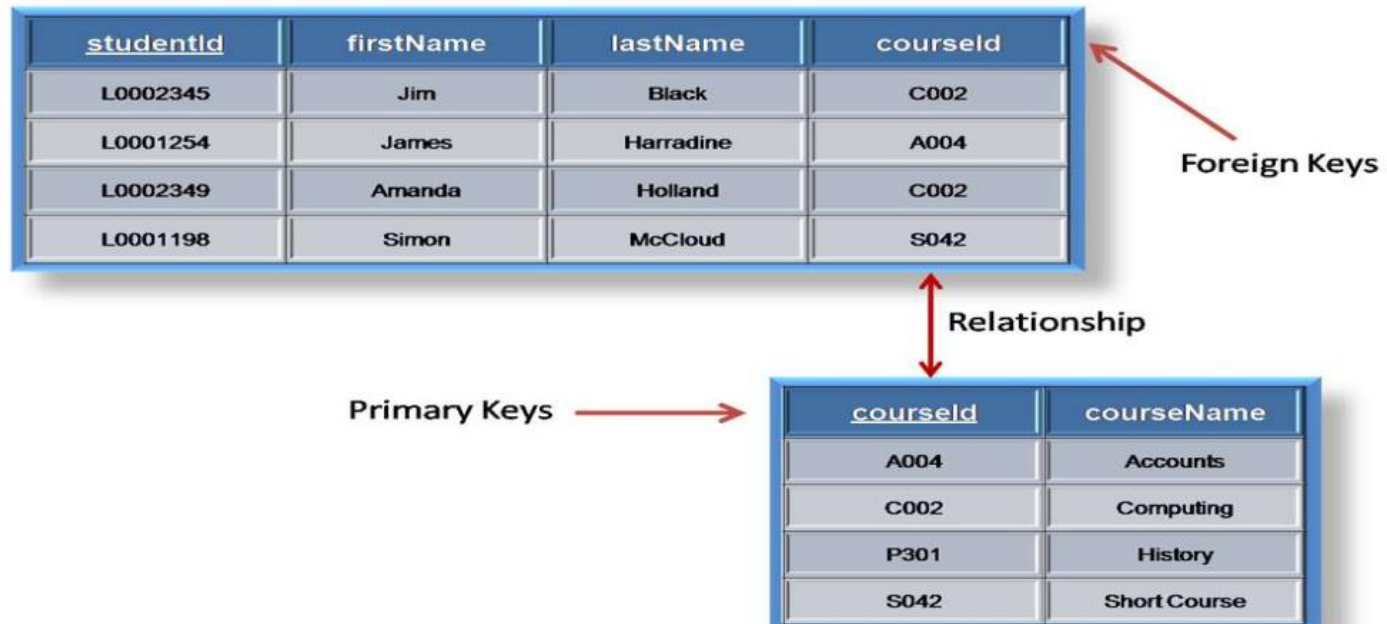


<u>StudentId</u>	firstName	lastName	courseId
L0002345	Jim	Black	C002
L0001254	James	Harradine	A004
L0002349	Amanda	Holland	C002
L0001198	Simon	McCloud	S042
L0023487	Peter	Murray	P301
L0018453	Anne	Norris	S042

Different types of keys

Foreign key:

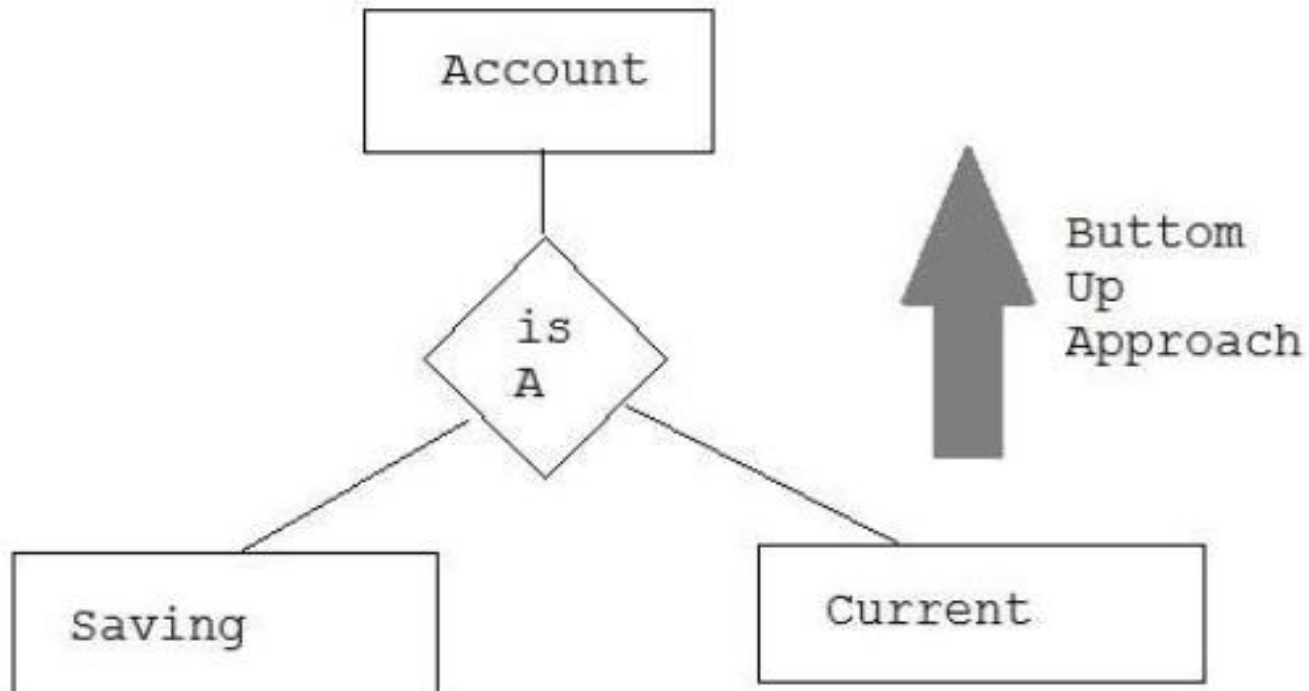
- A foreign key is generally a primary key from one table that appears as a field in another where the first table has a relationship to the second.
- In other words, if we had a table A with a primary key X, that linked to a table B where X was a field in B, then X would be a foreign key in B.



Extended ER features

Generalization

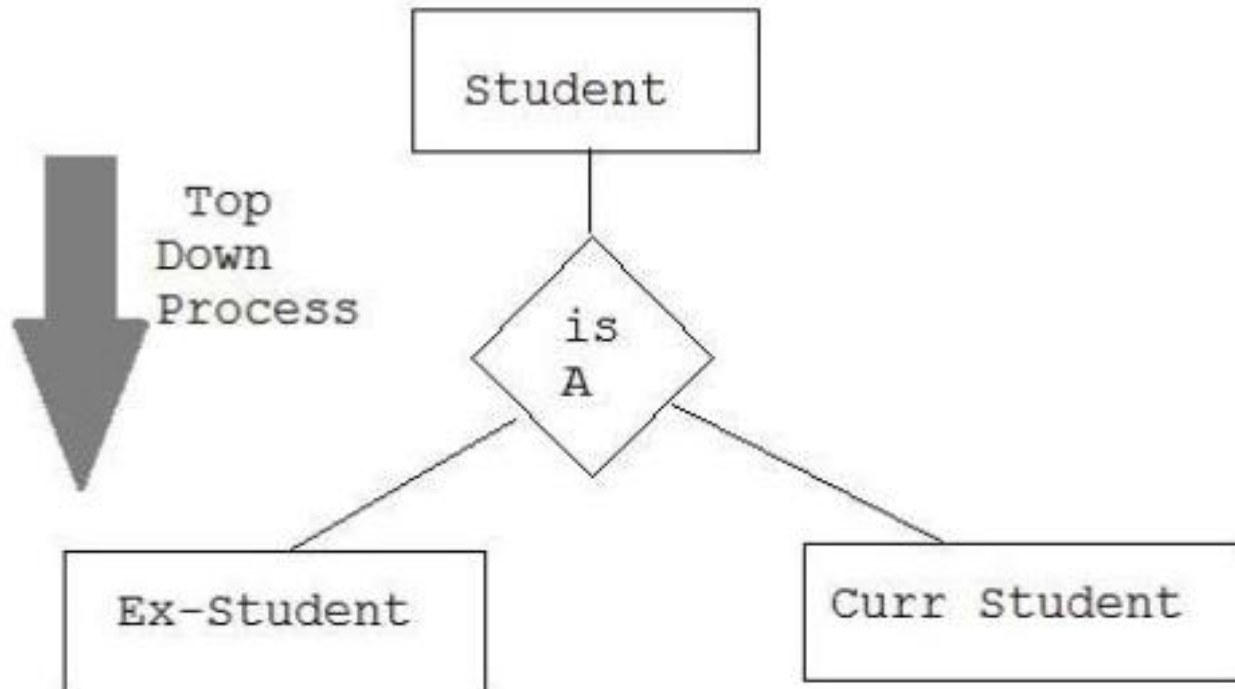
Generalization is a bottom-up approach in which two lower level entities combine to form a higher level entity. In generalization, the higher level entity can also combine with other lower level entity to make further higher level entity.



Extended ER features

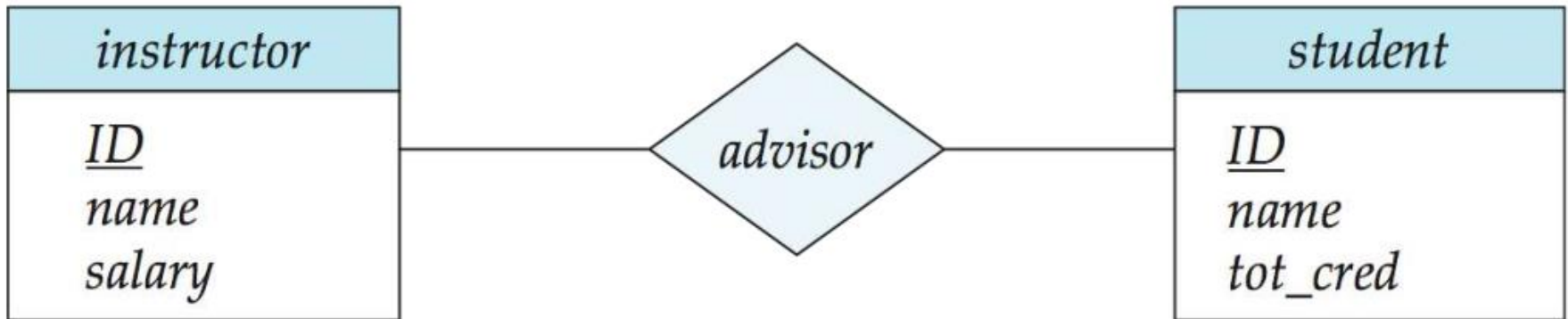
Specialization

Specialization is opposite to Generalization. It is a top-down approach in which one higher level entity can be broken down into two lower level entity. In specialization, some higher level entities may not have lower-level entity sets at all.



E-R Diagrams

- Rectangles represent entity sets.
- Diamonds represent relationship sets.
- Attributes listed inside entity rectangle. Or , as oval shape along with the rectangle.
- Underline indicates primary key attributes.



Summary of symbols used in E-R notation

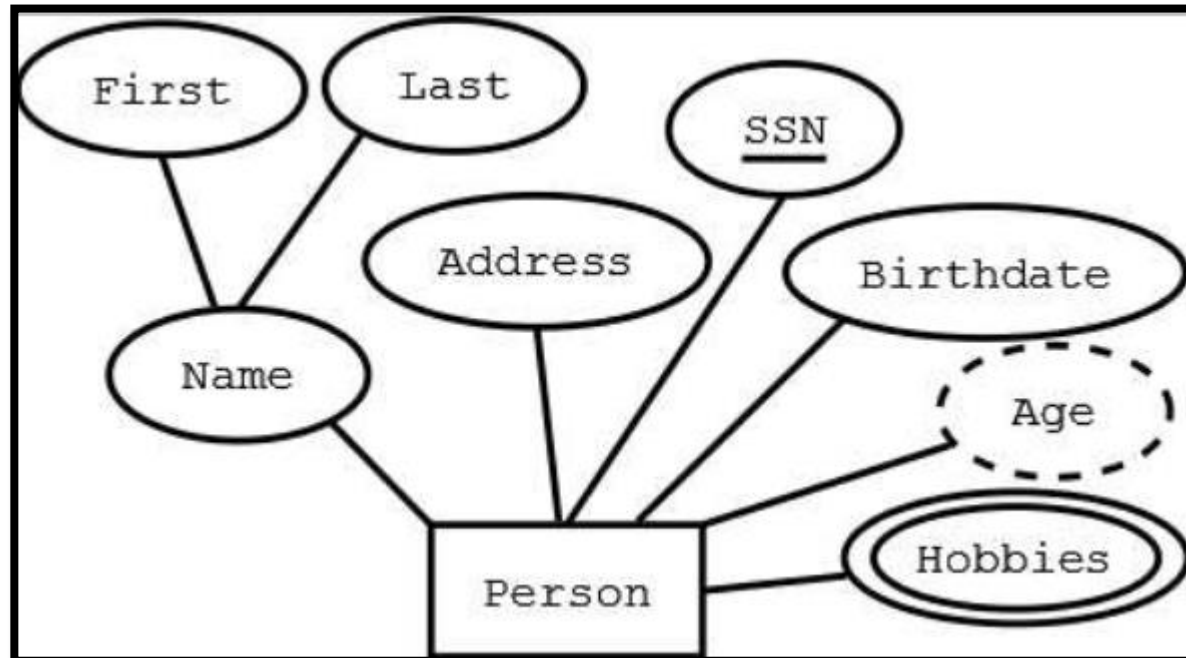
Representing attributes:

Rectangle: Entity

Ellipses: Attribute (underlined attributes are (part of) the primary key)

Double ellipses: multi-valued attribute

Dashed ellipses: derived attributes, e.g. age is derived from birthday and current date.



Thank you