International Islamic University Chittagong (IIUC)

Department of Computer and Communication Engineering

Program: B.sc (Eng.)
Course Code: CCE–3505
Type: Theory
Segment: Mid Term-Segment 2

Baizid MD Ashadzzaman
Designation: Adjunct Lecturer
Email: baizid.md.ashadzzaman@gmail.com
Phone: 8801862420119

Course Contents:

Contents	Note
Introduction to the Relational Model: Structure of Relational Databases, Database Schema, Keys, Schema, Diagrams, Relational Query Languages, Relational Operations.	

Entity-Relationship Model

Entity:

The Entity is a set of attributes in a database. An entity can be a real-world object which physically exists in this world. An entity may be any object, class, person or place. In other words, 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, and the values for some set of properties may uniquely identify an entity. For instance, a person may have a person id property whose value uniquely identifies that person.

Entity Set:

An entity set is a set of entities of the same type that share the same properties, or attributes. for example,The set of all people who are instructors at a university, can be defined as the entity set instructor. Similarly, the entity set student might represent the set of all students in the university.

76766	Crick
45565	Katz
10101	Srinivasan
98345	Kim
76543	Singh
22222	Einstein

instructor

98988	Tanaka
12345	Shankar
00128	Zhang
76543	Brown
76653	Aoi
23121	Chavez
44553	Peltier

student

Figure 7.1 Entity sets *instructor* and *student*.

A relationship is an association among several entities. For example, we can define a relationship advisor that associates instructor Katz with student Shankar. This relationship specifies that Katz is an advisor to student Shankar.

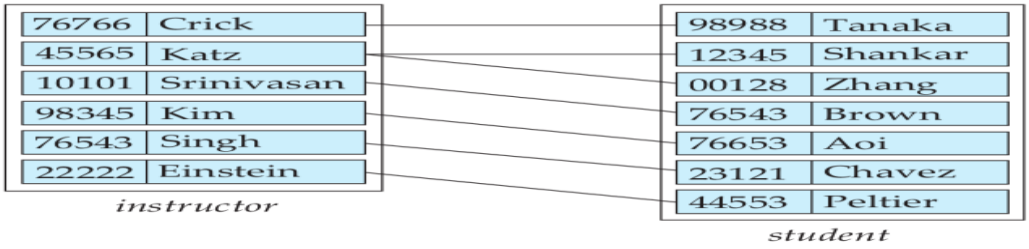


Figure 7.2 Relationship set *advisor*.

Relationship Set

A set of relationships of similar type is called a relationship set. Like entities, a relationship too can have attributes. These attributes are called descriptive attributes.

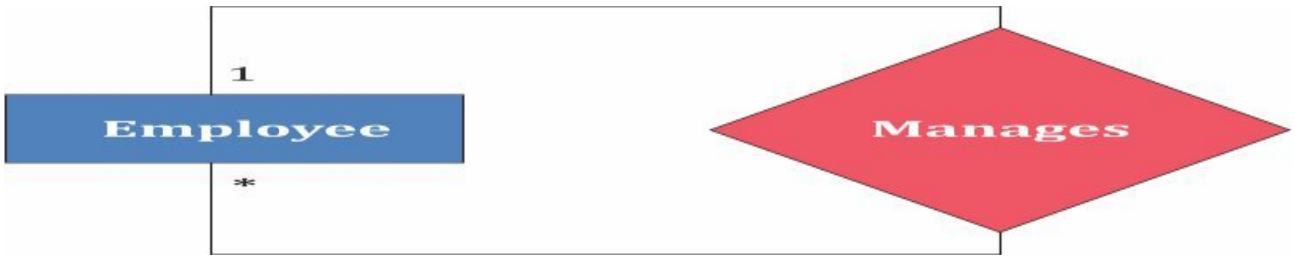
Types of Relationship Sets

A relationship set can be classified into the following types-

- Unary relationship set
- Binary relationship set
- Ternary relationship set

Unary Relationship Set

Unary relationship set is a relationship set where only one entity set participates in a relationship set.

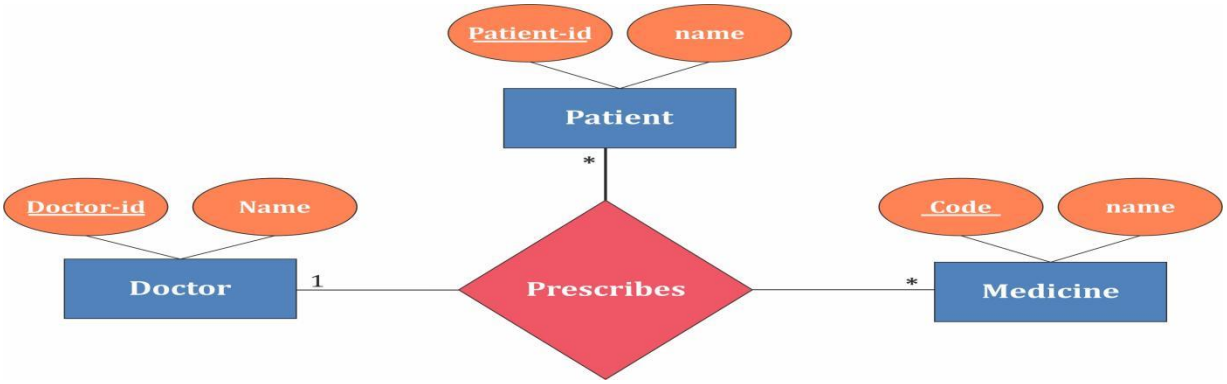


Binary Relationship Set

Binary relationship set is a relationship set where two entity sets participate in a relationship set.

Ternary Relationship Set

Ternary relationship set is a relationship set where three entity sets participate in a relationship set.



Mapping Cardinalities

Mapping cardinalities express the number of entities to which another entity can be associated via a relationship set.

For a binary relationship set R between entity sets A and B, the mapping cardinality must be one of the following:

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.

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.

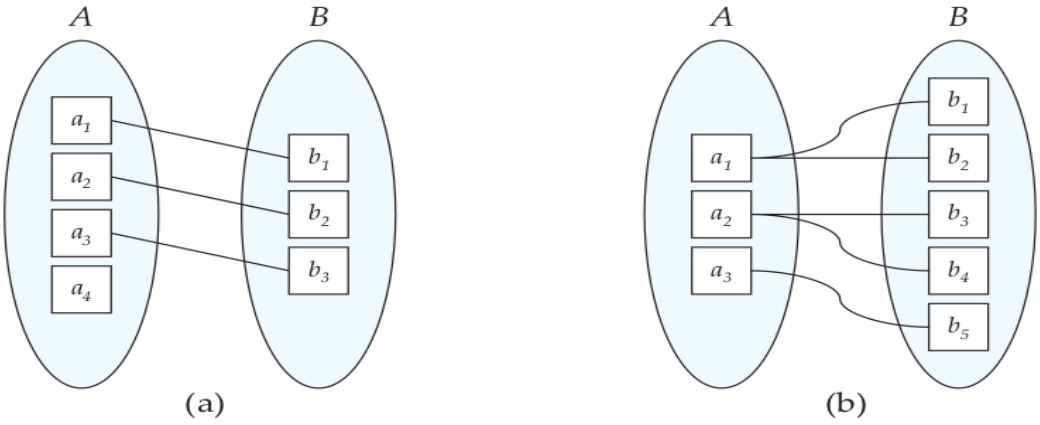


Figure 7.5 Mapping cardinalities. (a) One-to-one. (b) One-to-many.

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.

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.

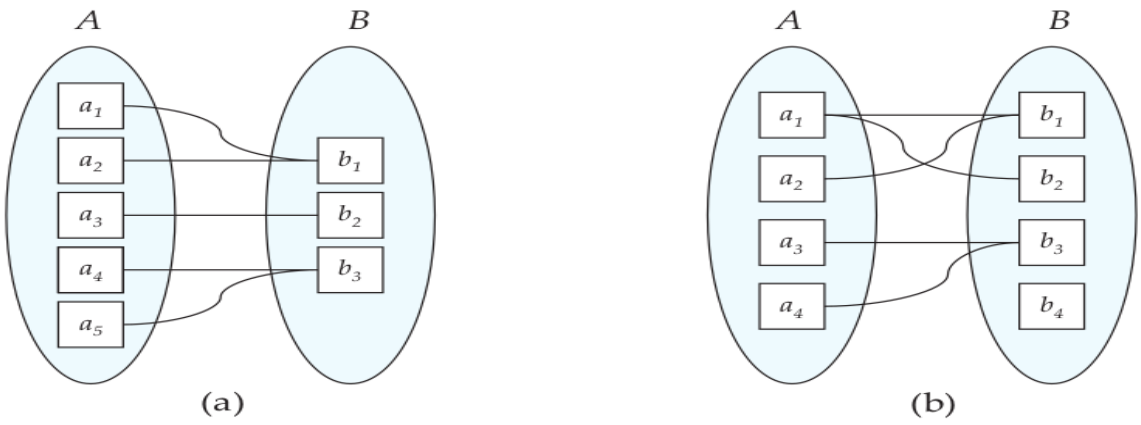


Figure 7.6 Mapping cardinalities. (a) Many-to-one. (b) Many-to-many.

Keys

A key in DBMS is an attribute or a set of attributes that help to uniquely identify a tuple (or row) in a relation (or table). Keys are also used to establish relationships between the different tables and columns of a relational database.

Different types of Keys:

Super key: A superkey is a set of one or more attributes that collectively allow us to uniquely identify a tuple in the relation. Super key attributes can also contain NULL values.

(class, section, roll) => superkey

(class, section, roll, gender) => superkey

Candidate key: Minimal superkeys are called candidate keys. Candidate key attributes can also contain NULL values.

(class, section, roll) => candidate key

Primary key: A primary key is a column in a relational database table that is unique for each record. It doesn't permit null values.

(id, class, section, name) where id is the primary key.

Composite key: A composite Key is a set of two or more attributes that help identify each tuple in a table uniquely.

(class, section, roll) => composite key

Foreign key: A foreign key is a field in one table, that is used as the primary key in another table. Foreign Key is used to establish relationships between two tables.

Attribute

In a database management system (DBMS), an attribute is a piece of data that describes an entity.In other words, an attribute is the characteristics of an entity. For example, in a customer database, the attributes might be name, address, and phone number. In a product database, the attributes might be name, price, and date of manufacture.

Types of Attributes

Simple attributes: Simple attributes are those that cannot be further divided into sub-attributes. For example, A student's roll number of a student or the employee identification number.

Composite attributes: Composite attributes are made up of two or more simple attributes. For example, a person's address may be a composite attribute that is made up of the person's street address, city, state, and zip code.

Single-Valued Attributes: Single-valued attributes can only have one value. For example, a person's NID number or DOB is a single-valued attribute.

Multivalued Attributes: Multivalued attributes can have more than one value. For example, a person may have multiple email addresses or phone numbers.

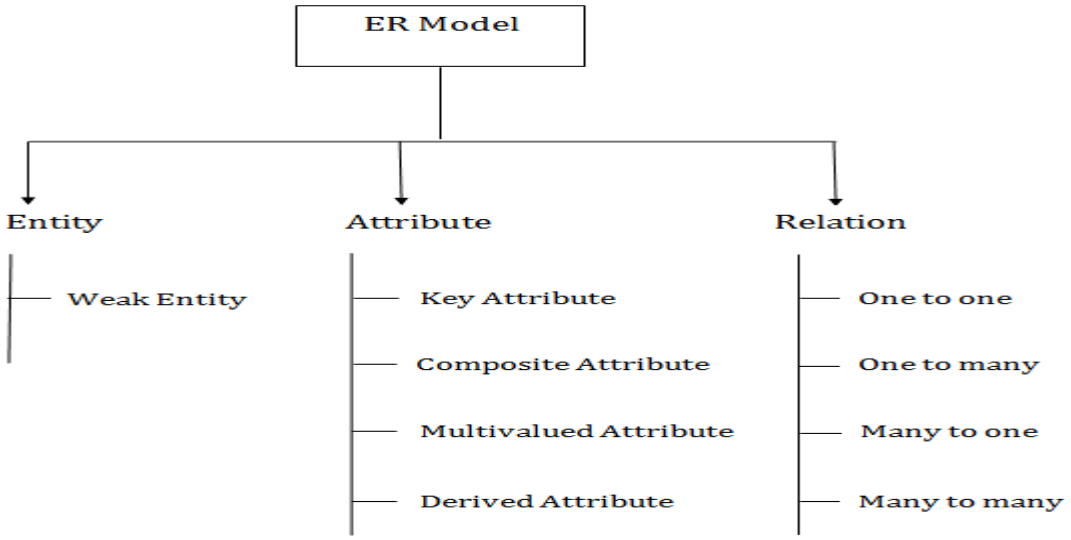
Derived attributes: Derived attributes are based on other attributes and are not stored directly in the database. For example: Consider a database of employees. Each employee has a date of birth, and we might want to calculate their age. However, age is a derived attribute because it can be determined from the date of birth.

ER or Entity Relationship Diagram in DBMS

ER model stands for an Entity-Relationship model. It is a high-level data model. This model is used to define the data elements and relationship for a specified system.

It develops a conceptual design for the database. It also develops a very simple and easy to design view of data.

Component of ER Diagram



1. Entity:

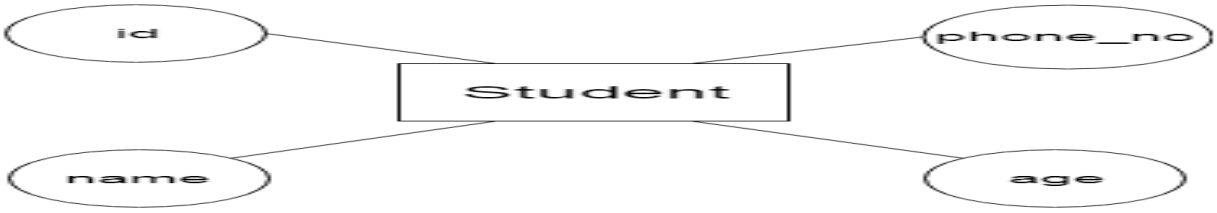
An entity may be any object, class, person or place. In the ER diagram, an entity can be represented as rectangles. Consider an organization as an example- manager, product, employee, department etc. can be taken as an entity.

Weak Entity

An entity that depends on another entity called a weak entity. The weak entity doesn't contain any key attribute of its own. The weak entity is represented by a double rectangle.

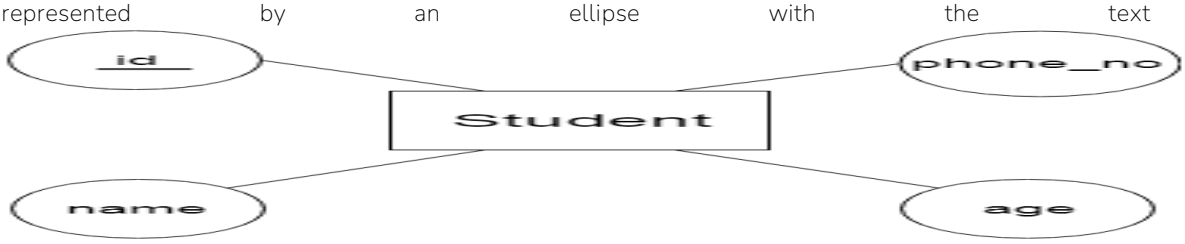
2. Attribute

The attribute is used to describe the property of an entity. Eclipse is used to represent an attribute. For example, id, age, contact number, name, etc. can be attributes of a student.



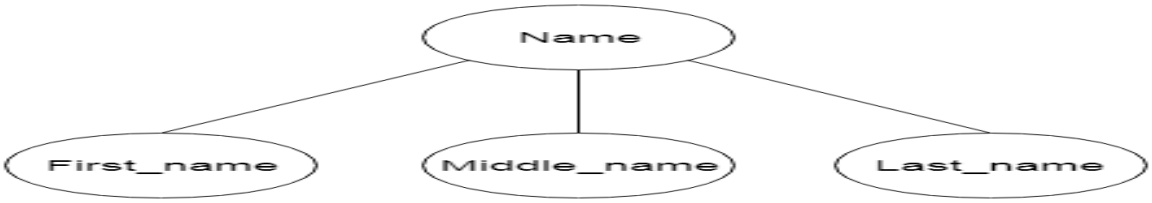
Key Attribute

The key attribute is used to represent the main characteristics of an entity. It represents a primary key. The key attribute is represented by an ellipse with the text underlined.



Composite Attribute

An attribute that composed of many other attributes is known as a composite attribute. The composite attribute is represented by an ellipse, and those ellipses are connected with an ellipse.

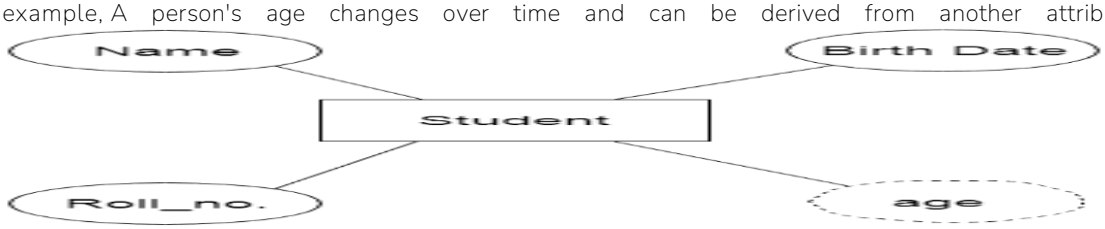


Multivalued Attribute

An attribute can have more than one value. These attributes are known as a multivalued attribute. The double oval is used to represent multivalued attribute. For example, a student can have more than one phone number.

Derived Attribute

An attribute that can be derived from other attribute is known as a derived attribute. It can be represented by a dashed ellipse. For example, A person's age changes over time and can be derived from another attribute like Date of birth.



3. Relationship

A relationship is used to describe the relation between entities. Diamond is used to represent the relationship.

Types of relationship are as follows:

a. One-to-One Relationship

A relation is said to be One to One relationship between two tables in which a single row of the first table can only be related to one and only one records of a second table. For example, A female can marry to one male, and a male can marry to one female.

b. One-to-many relationship

A relation is said to be One-to-many relationship between two tables in which a single rows of the first table can be related to one or more rows of the second tables. For example, Scientist can invent many inventions, but the invention is done by the only specific scientist.

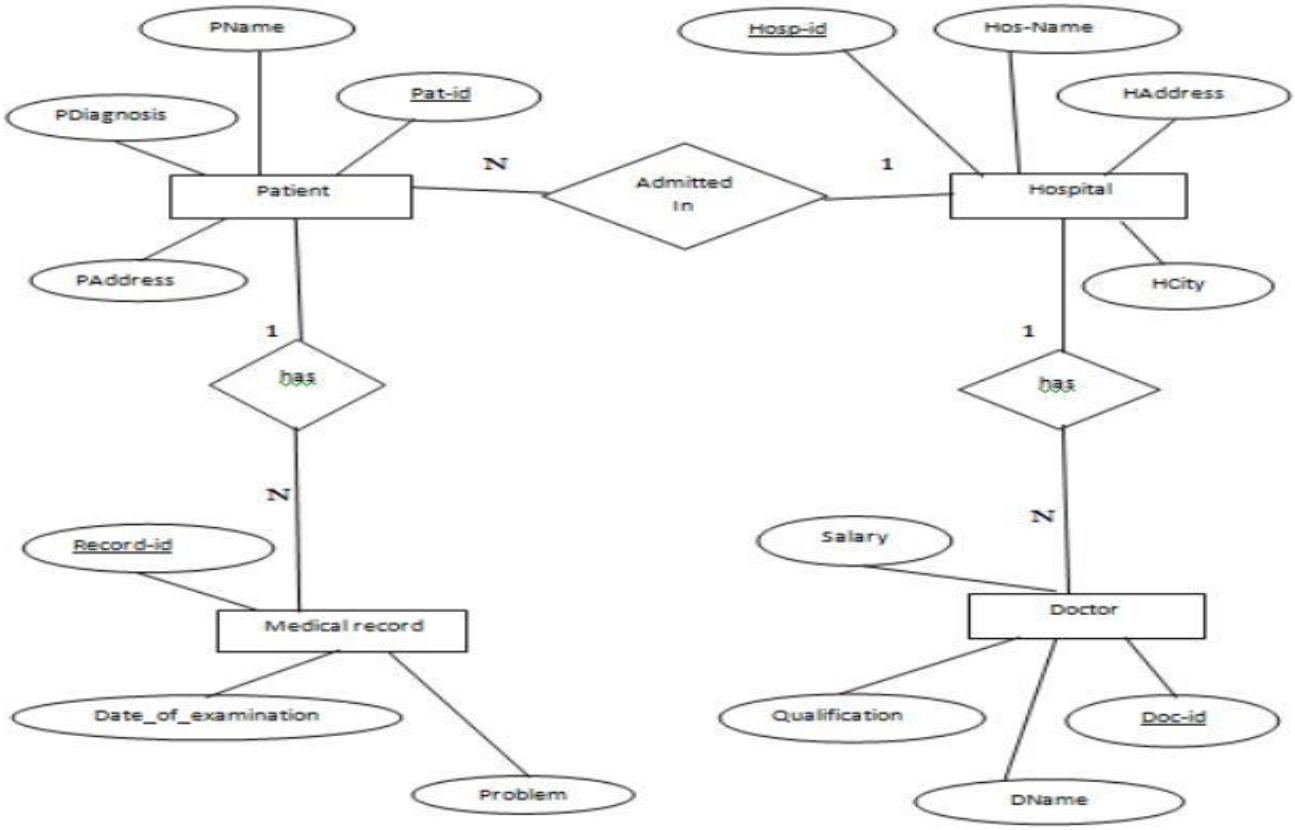
c. Many-to-one relationship

A relation is said to be Many-to-one relationship between two tables in which each rows of the first table can be related to only one row of the second tables. For example, Student enrolls for only one course, but a course can have many students.

d. Many-to-many relationship

A relation is said to be Many-to-many relationship between two tables in which Each record of the first table can be related to any records in the second table. For example, Employee can assign by many projects and project can have many employees.

Example of R Diagram

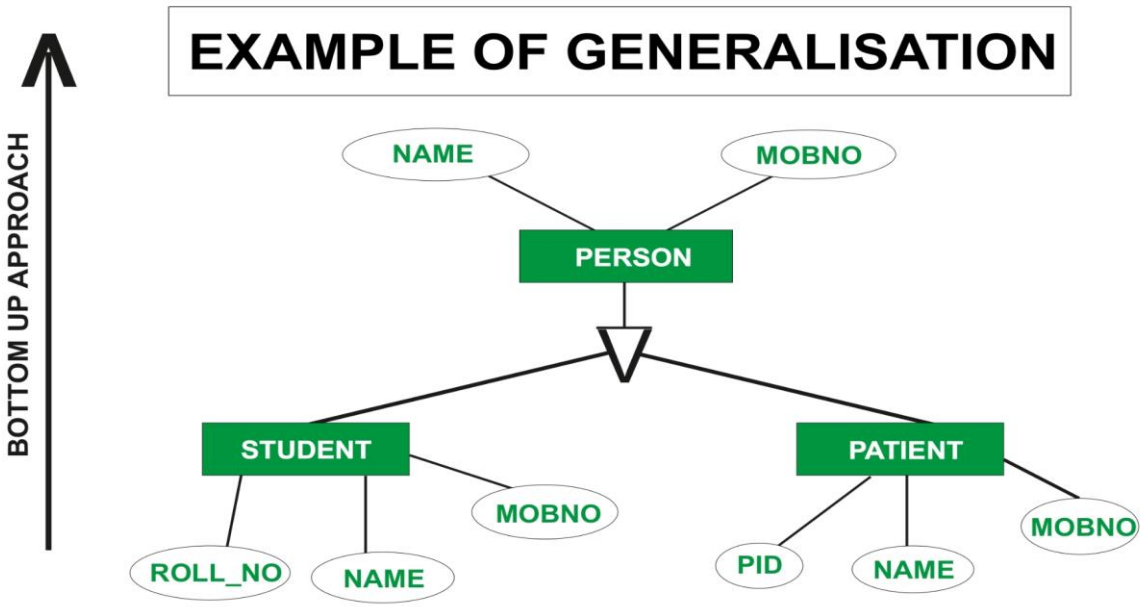


Generalization:

Generalization is a bottom-up approach in which two or more entities of lower level are combined to form a higher level entity if they have some attributes in common It works on the principle of bottom-up approach.

In Generalization lower-level functions are combined to form higher-level function which is called entities. This process is repeated further to make advanced-level entities. In the Generalization process properties are drawn from particular entities and thus we can create generalized entities. We can summarize the Generalization process as it combines subclasses to form a superclass.

Example of Generalization – Consider two entities Student and Patient. These two entities will have some characteristics of their own. For example, Student entity will have Roll_No, Name and Mob_No while patient will have PId, Name and Mob_No characteristics. Now in this example Name and Mob_No of both Student and Patient can be combined as a Person to form one higher-level entity and this process is called as Generalization Process.



Specialization

Specialization is a top-down approach in which one higher level entity can be broken down into two lower level entities. Specialization is used to identify the subset of an entity that shares some distinguishing characteristics.

Example of Specialization – Consider an entity Account. This will have some attributes

consider them Acc_No and Balance. Account entity may have some other attributes like

Current_Acc and Savings_Acc. Now Current_Acc may have Acc_No, Balance and

Transactions while Savings_Acc may have Acc_No, Balance and Interest_Rate. Hence we can say that specialized entities inherits characteristics of higher level entity.