Chapter 2
The Entity- Relationship Model

IT – SE – Database Management Systems

Savita Lohiya
Assistant professor
Dept.of Information Technology
SIES Graduate School of Technology

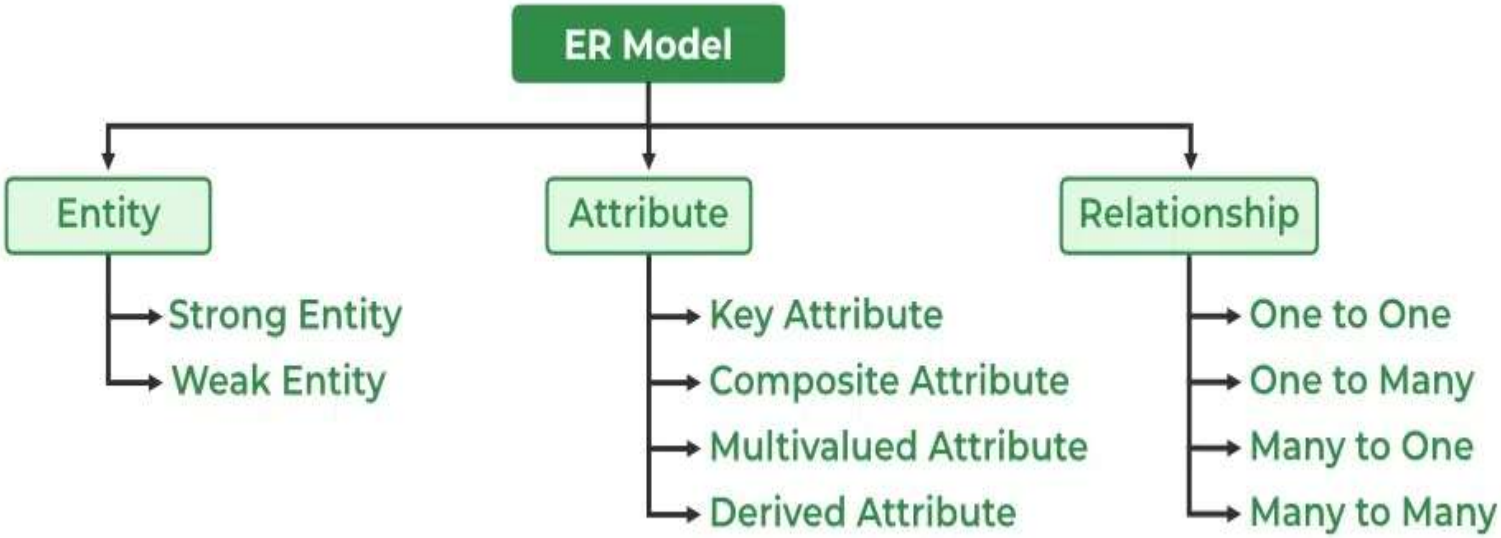
Database Management System (DBMS)

The Entity– Relationship Model
Conceptual Modeling of a database : The Entity– Relationship (ER) Model, Entity Type, Entity Sets, Attribute Types and Keys, Relationship Types, Relationship Sets, Weak entity Types.

ER model –– Database Modeling

- The Entity–Relationship Model (ER Model) is a conceptual model for designing a databases. This model represents the logical structure of a database, including entities, their attributes and relationships between them.
- **Entity:** An objects that is stored as data such as *Student*, *Course* or *Company*.
- **Attribute:** Properties that describes an entity such as *StudentID*, *CourseName*, or *EmployeeEmail*.
- **Relationship:** A connection between entities such as "a *Student* enrolls in a *Course*".

Entity Relationship model/diagram



Entity Relationship model/diagram

ER Model in Database Design Process

- We typically follow the below steps for designing a database for an application.

- Gather the requirements (functional and data) by asking questions to the database users.
- Create a logical or conceptual design of the database. This is where ER model plays a role. It is the most used graphical representation of the conceptual design of a database.

Entity Relationship model/diagram

- **Why Use ER Diagrams In DBMS?**

- ER diagrams represent the E-R model in a database, making them easy to convert into relations (tables).
- These diagrams serve the purpose of real-world modeling of objects which makes them intently useful.
- ER diagrams require no technical knowledge of the underlying DBMS used.
- They visually model data and its relationships, making complex systems easier to understand.

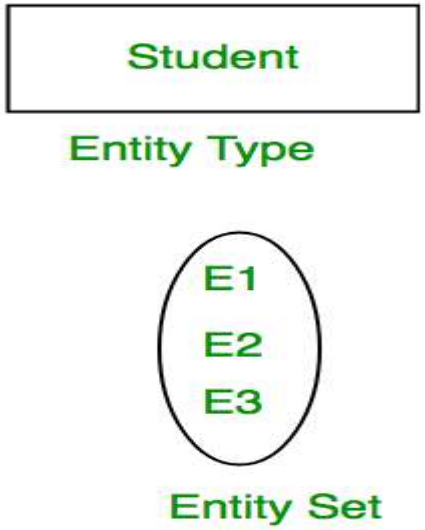
What is an Entity?

- An Entity represents a real-world object, concept or thing about which data is stored in a database.
- It act as a building block of a database. Tables in relational database represent these entities.

- Example of entities:
- **Real-World Objects:** Person, Car, Employee etc.
- **Concepts:** Course, Event, Reservation etc.
- **Things:** Product, Document, Device etc.

Entity Relationship model/diagram

- **What is an Entity Set?**
- An entity refers to an individual object of an entity type, and the collection of all entities of a particular type is called an entity set. For example, E1 is an entity that belongs to the entity type "Student," and the group of all students forms the entity set.



Types of Entity

- There are two main types of entities:
- **1. Strong Entity**
- It is a type of entity that has a key Attribute that can uniquely identify each instance of the entity.
- A Strong Entity does not depend on any other Entity in the Schema for its identification.
- It has a primary key that ensures its uniqueness and is represented by a rectangle in an ER diagram.

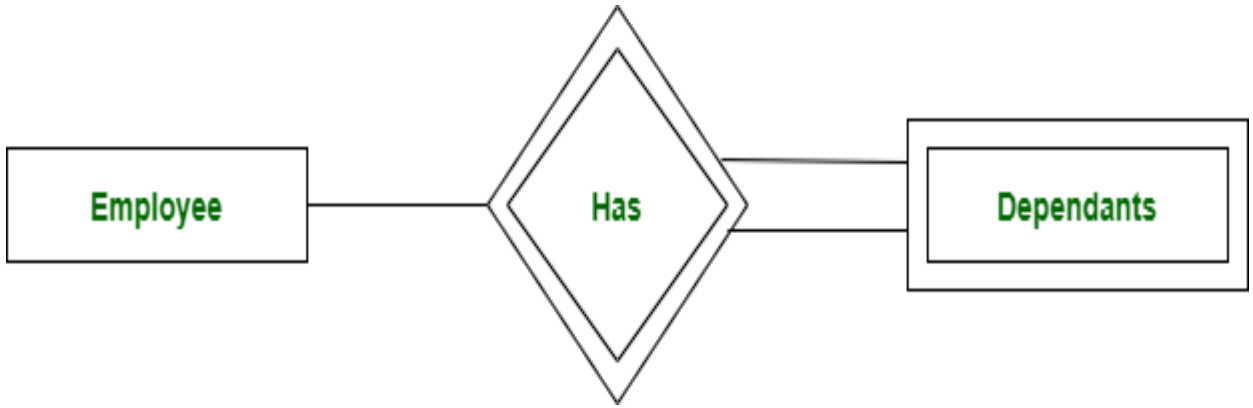
Types of Entity

- **2. Weak Entity**
- A Weak Entity cannot be uniquely identified by its own attributes alone.
- It depends on a strong entity to be identified. A weak entity is associated with an identifying entity (strong entity), which helps in its identification.
- A weak entity are represented by a double rectangle. The participation of weak entity types is always total.

- The relationship between the weak entity type and its identifying strong entity type is called identifying relationship and it is represented by a double diamond.

Types of Entity

- **2. Weak Entity**
- **Example:**
- A company may store the information of dependents (Parents, Children, Spouse) of an Employee.
- But the dependents can't exist without the employee. So dependent will be a Weak Entity Type and Employee will be identifying entity type for dependent, which means it is Strong Entity Type.



Types of Attributes

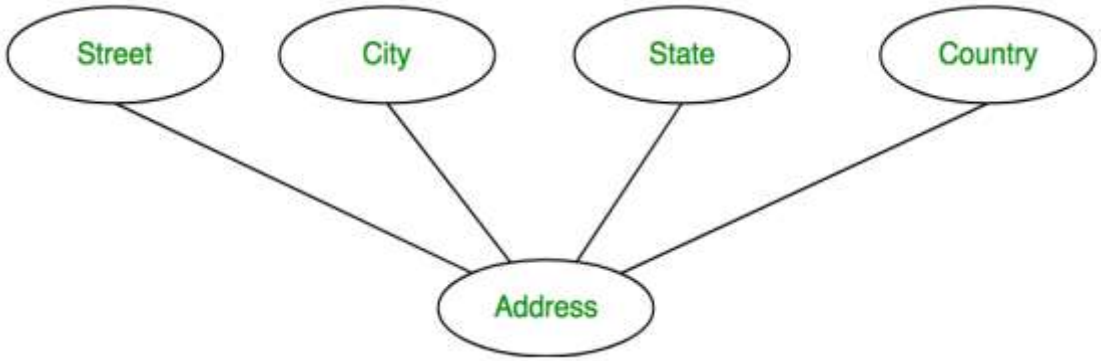
- **1. Key Attribute**

  The attribute which uniquely identifies each entity in the entity set is called the key attribute.

  For example, Roll_No will be unique for each student. In ER diagram, the key attribute is represented by an oval with an underline.

Types of Attributes

- **2. Composite Attribute**
- An attribute composed of many other attributes is called a composite attribute.
- For example, the Address attribute of the student Entity type consists of Street, City, State, and Country.
- In ER diagram, the composite attribute is represented by an oval comprising of ovals.



Types of Attributes

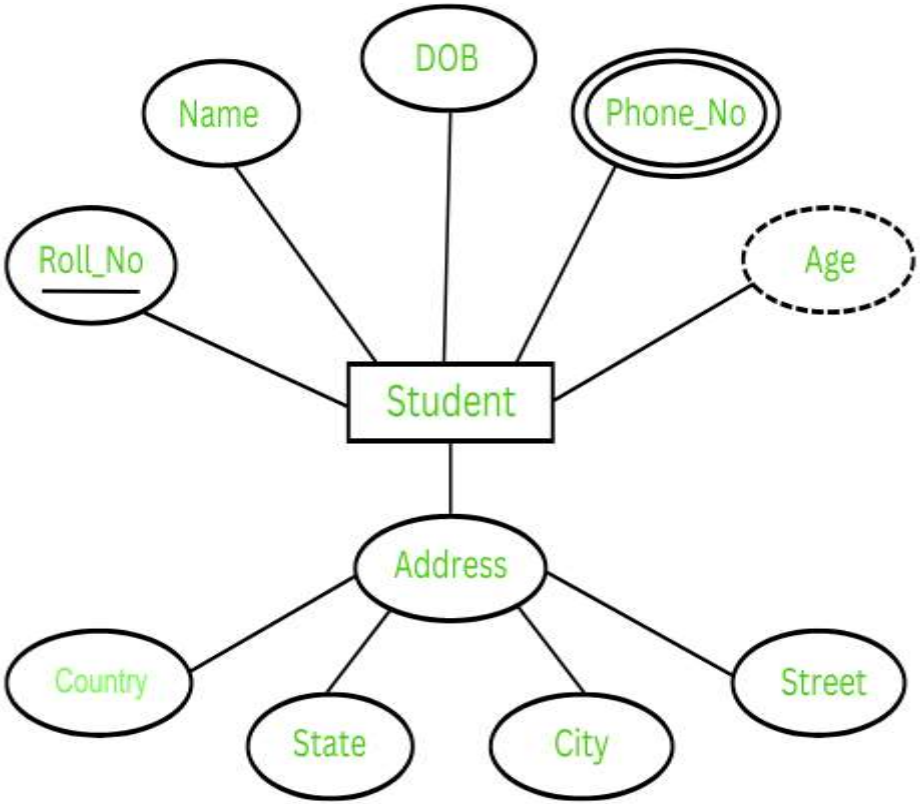
- **3. Multivalued Attribute**
- An attribute consisting of more than one value for a given entity. For example, Phone_No (can be more than one for a given student).
- In ER diagram, a multivalued attribute is represented by a double oval.

Types of Attributes

- **4. Derived Attribute**
- An attribute that can be derived from other attributes of the entity type is known as a derived attribute. e.g.; Age (can be derived from DOB). In ER diagram, the derived attribute is represented by a dashed oval.

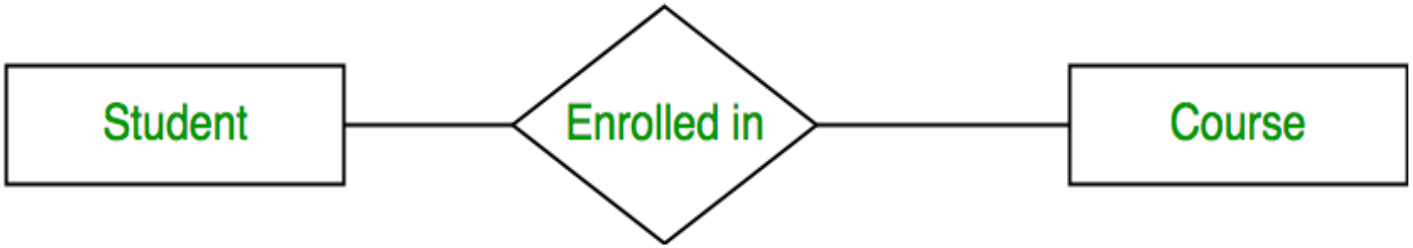
Types of Attributes

- The Complete Entity Type Student with its Attributes can be represented as:



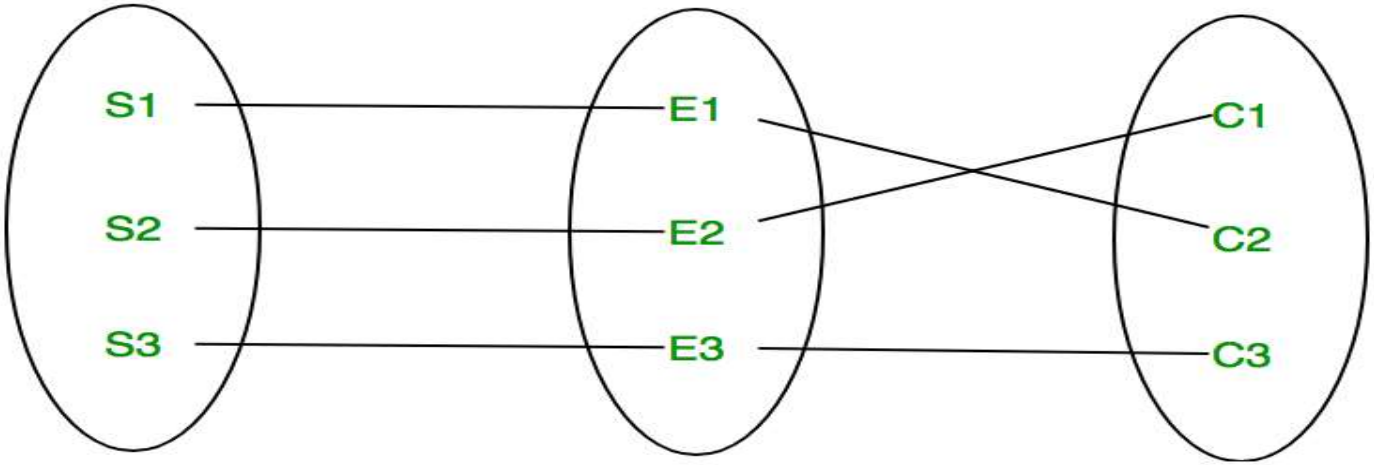
Relationship Type and Relationship Set

- **Relationship Type and Relationship Set**
- A Relationship Type represents the association between entity types. For example, 'Enrolled in' is a relationship type that exists between entity type Student and Course.
- In ER diagram, the relationship type is represented by a diamond and connecting the entities with lines.



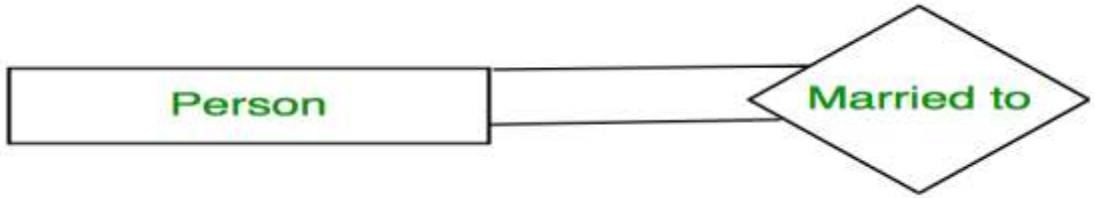
Relationship Type and Relationship Set

- A set of relationships of the same type is known as a relationship set. The following relationship set depicts S1 as enrolled in C2, S2 as enrolled in C1, and S3 as registered in C3.



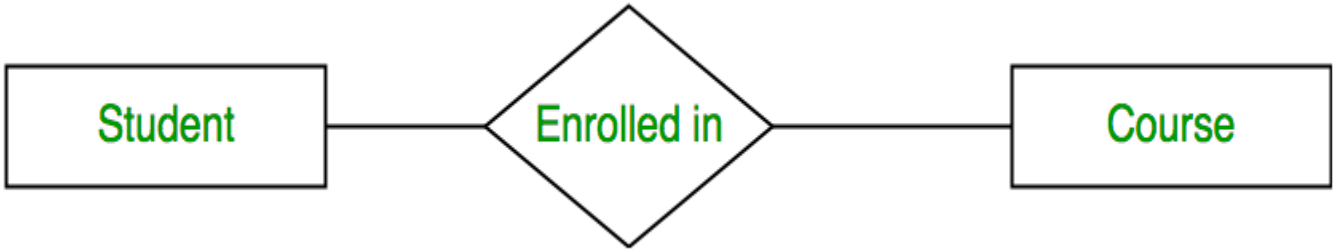
Degree of a Relationship Set

- The number of different entity sets participating in a relationship set is called the degree of a relationship set.
- **1. Unary Relationship:** When there is only ONE entity set participating in a relation, the relationship is called a unary relationship. For example, one person is married to only one person.



Degree of a Relationship Set

- **2. Binary Relationship:** When there are TWO entities set participating in a relationship, the relationship is called a binary relationship. For example, a Student is enrolled in a Course.
-



Degree of a Relationship Set

- **3. Ternary Relationship:** When there are three entity sets participating in a relationship, the relationship is called a ternary relationship.

- **4. N-ary Relationship:** When there are n entities set participating in a relationship, the relationship is called an n-ary relationship.

Cardinality in ER Model

- The maximum number of times an entity of an entity set participates in a relationship set is known as_cardinality.
- Cardinality can be of different types:
- **1. One–to–One**
- When each entity in each entity set can take part only once in the relationship, the cardinality is one–to–one.
- Let us assume that a male can marry one female and a female can marry one male. So the relationship will be one–to–one.

.

Cardinality in ER Model

Cardinality in ER Model

2. One-to-Many

- In one-to-many mapping as well where each entity can be related to more than one entity.
- Let us assume that one surgeon department can accommodate many doctors.
- So the Cardinality will be 1 to M. It means one department has many Doctors.

Cardinality in ER Model

Cardinality in ER Model

4. Many–to–Many

- When entities in all entity sets can take part more than once in the relationship cardinality is many to many.
- Let us assume that a student can take more than one course and one course can be taken by many students. So the relationship will be many to many.

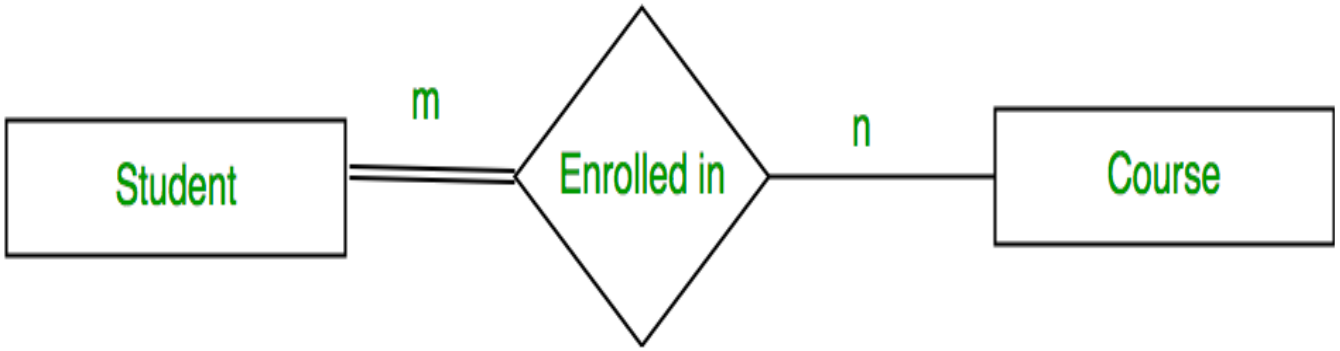
Participation Constraint

- Participation Constraint is applied to the entity participating in the relationship set.
- **1. Total Participation:** Each entity in the entity set must participate in the relationship. If each student must enroll in a course, the participation of students will be total. Total participation is shown by a double line in the ER diagram.

- **2. Partial Participation:** The entity in the entity set may or may NOT participate in the relationship. If some courses are not enrolled by any of the students, the participation in the course will be partial.

- The diagram depicts the 'Enrolled in' relationship set with Student Entity set having total participation and Course Entity set having partial participation.

Participation Constraint

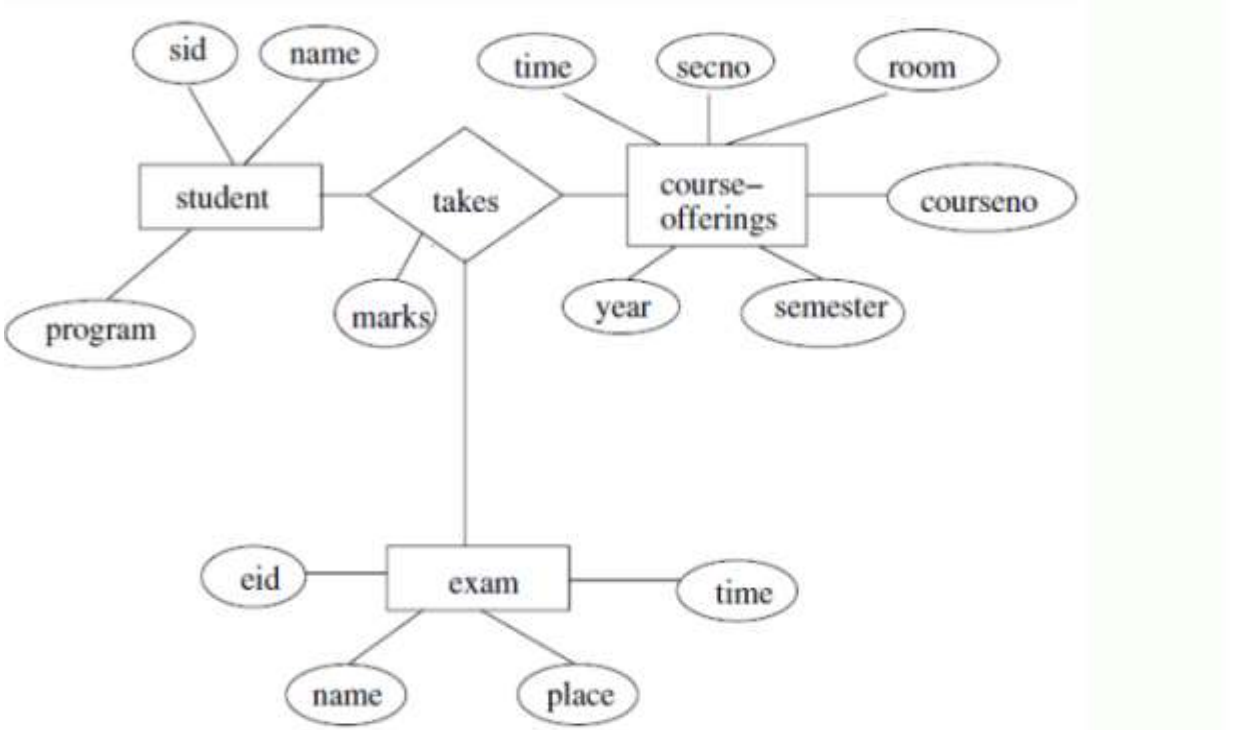


E-R Diagram case study

- Consider a database used to record the marks that students get in different exams of different course offerings.

- a) Construct an E-R diagram that models exams as entities, and uses a ternary relationship, for the above database.

E-R Diagram case study



Keys in E-R Diagram

- 1.Primary Key:
- A primary key is a unique key, meaning it can uniquely identify each record (tuple) in a table.
- It must have unique values and cannot contain any duplicate values.
- A primary key cannot be NULL, as it needs to provide a valid, unique identifier for every record.
- For Example, STUD_NO

Keys in E-R Diagram

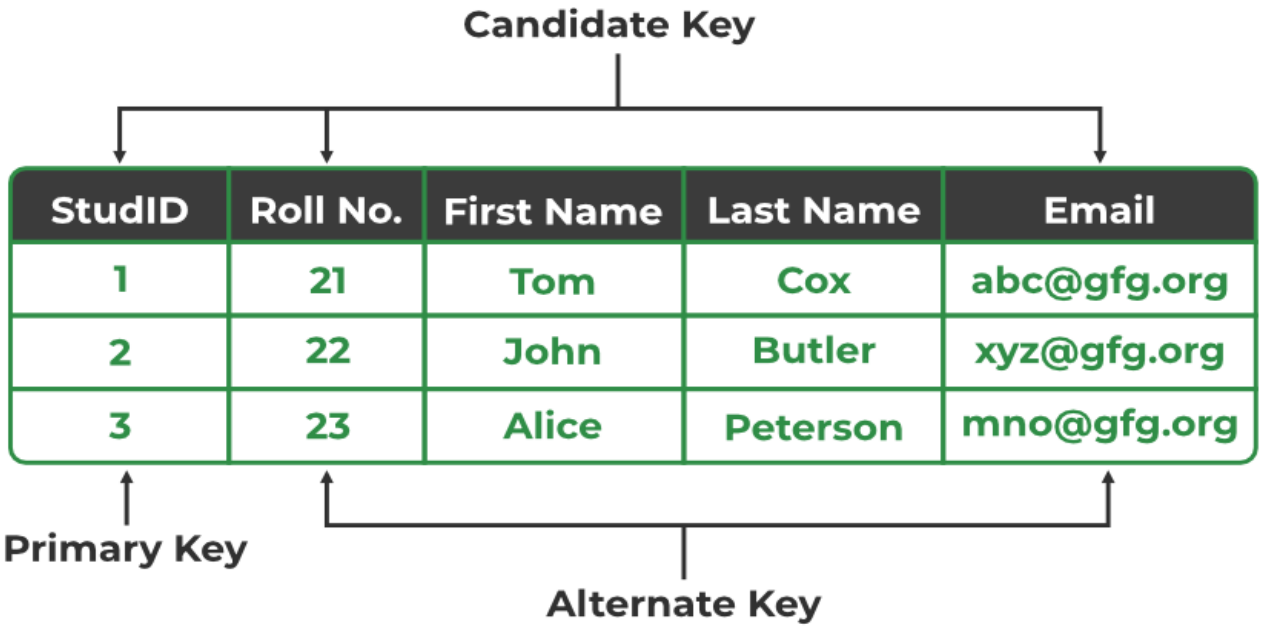
- 2. **Super Key**
- The set of one or more attributes (columns) that can uniquely identify a tuple (record) is known as Super key
- It may include extra attributes that aren't important for uniqueness but still uniquely identify the row. For Example, STUD_NO, (STUD_NO, STUD_NAME), etc.
- A super key is a group of single or multiple keys that uniquely identifies rows in a table. It supports NULL values in rows.

- For example, if the "STUD_NO" column can uniquely identify a student, adding "SNAME" to it will still form a valid super key, though it's unnecessary.

Keys in E–R Diagram

- **3.Candidate Key**
- The minimal set of attributes that can uniquely identify a tuple is known as a candidate key.
- For Example, STUD_NO in STUDENT relation.
- A candidate key is a minimal super key, meaning it can uniquely identify a record but contains no extra attributes.
- A candidate key must contain unique values, ensuring that no two rows have the same value in the candidate key's columns.
- Every table must have at least a single candidate key.
- A table can have multiple candidate keys but only one primary key.

Keys in E-R Diagram



Keys in E–R Diagram

4. Alternate Key

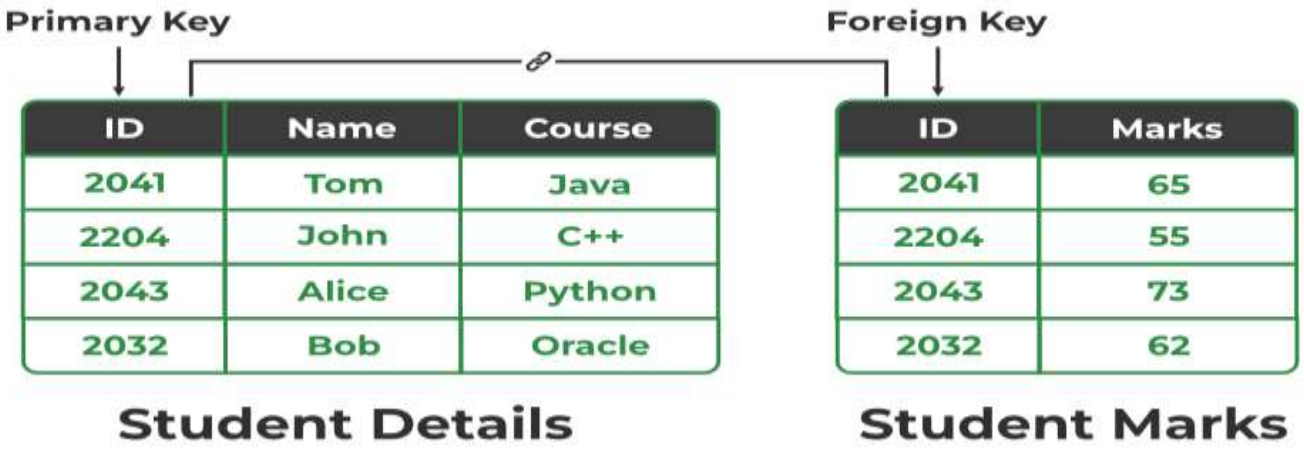
- An alternate key is any candidate key in a table that is not chosen as the primary key.
- In other words, all the keys that are not selected as the primary key are considered alternate keys.

- An alternate key is also referred to as a secondary key because it can uniquely identify records in a table, just like the primary key.

- An alternate key can consist of one or more columns (fields) that can uniquely identify a record, but it is not the primary key
- **Example:** In the STUDENT table, both STUD_NO and PHONE are candidate keys. If STUD_NO is chosen as the primary key, then PHONE would be considered an alternate key.

Keys in E–R Diagram

- **5. Foreign Key**
- A foreign key is an attribute in one table that refers to the primary key in another table.
- The table that contains the foreign key is called the referencing table and the table that is referenced is called the referenced table.
- .



Keys in E-R Diagram

- A foreign key in one table points to the primary key in another table, establishing a relationship between them.
- It helps connect two or more tables, enabling you to create relationships between them. This is important for maintaining data integrity and preventing data redundancy.
- They act as a cross-reference between the tables.
- Foreign Key can be NULL .

Keys in E-R Diagram

- **6. Composite Key**
- Sometimes, a table might not have a single column/attribute that uniquely identifies all the records of a table.
- To uniquely identify rows of a table, a combination of two or more columns/attributes can be used
- It acts as a primary key if there is no primary key in a table
- Two or more attributes are used together to make a composite key .
- Different combinations of attributes may give different accuracy in terms of identifying the rows uniquely.

- **Example:** In the STUDENT_COURSE table, {STUD_NO, COURSE_NO} can form a composite key to uniquely identify each record.

Keys in E–R Diagram

- **Composite key**

STUD_NO	TEACHER_NO	COURSE_NO
1	005	C001
2	056	C005

EER Diagram((Extended Entity-Relationship)

- An EER (Enhanced Entity-Relationship) diagram is a visual representation used in database design that extends the basic Entity-Relationship (ER) model to handle more complex data relationships.

- It incorporates concepts like generalization, specialization, and aggregation to model real-world scenarios more accurately and efficiently.

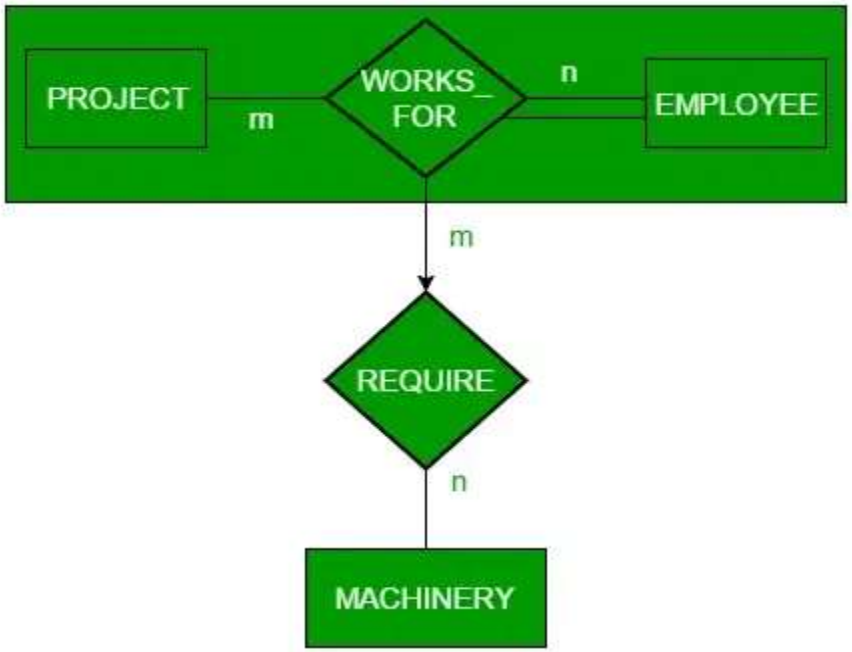
EER Diagram((Extended Entity-Relationship)

- Key Concepts in EER Diagrams:
- **Generalization/Specialization:** Allows grouping of similar entities into supertypes and subtypes. For example, an "Employee" entity can be generalized into subtypes like "Manager", "Engineer", and "Analyst", each with their own specific attributes.
- **Aggregation:** Represents a "whole-part" relationship between entities. For instance, an "Engine" can be part of a "Car".
- **Inheritance:** Subtypes inherit the attributes and relationships of their supertypes.
- Category or Union type
- Attribute and Relationship

EER Diagram((Extended Entity-Relationship)

- **Aggregation**
- An ER diagram is not capable of representing higher relationship
- In those cases, a relationship with its corresponding entities is aggregated into a higher-level entity.
- Aggregation is an abstraction through which we can represent relationships as higher-level entity sets.

EER Diagram((Enhanced Entity-Relationship)



Aggregation

EER Diagram((Enhanced Entity-Relationship)

- *Aggregation is also called as "Higher-Order Relationship".*

EER Diagram((Extended Entity-Relationship)

- **Union or category**
- It brings together entities from different, potentially unrelated, entity types.
- **Example:**
- Consider a scenario where a "Car Owner" could be a "Person", a "Bank", or a "Company". The "Car Owner" category is a subclass of the union of these three superclasses.

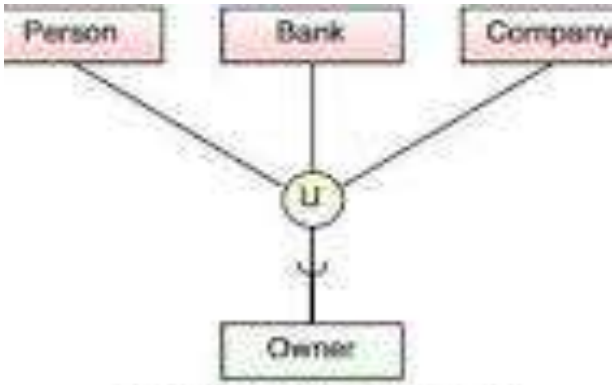
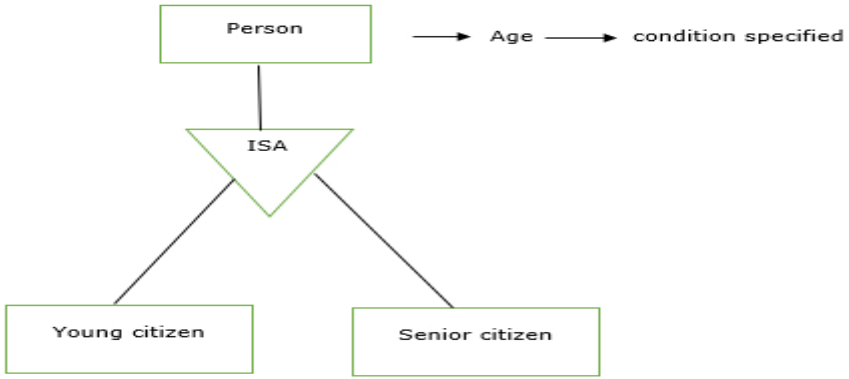


Fig. Categories (Union Type)

Constraints on the generalization and specialization

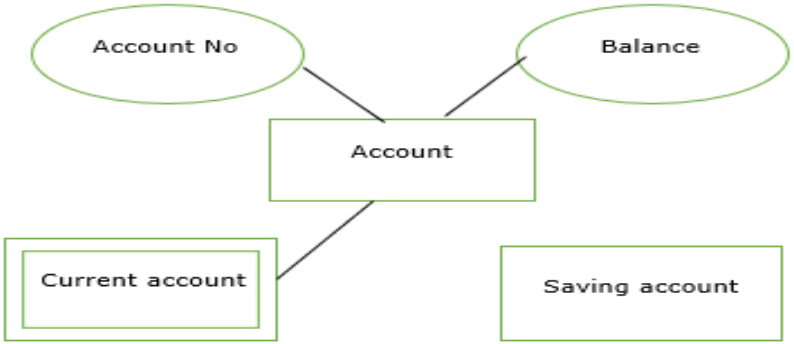
- 1.Conditional definition
- Create one database, and keep conditions on one attribute for example attendance. This type of constraint is defined on a single attribute which is further dividing an entity into two sub entity sets which will give information for the given attribute.
- Example
- Given below is an example of the conditional definition database –



Constraints on the generalization and specialization

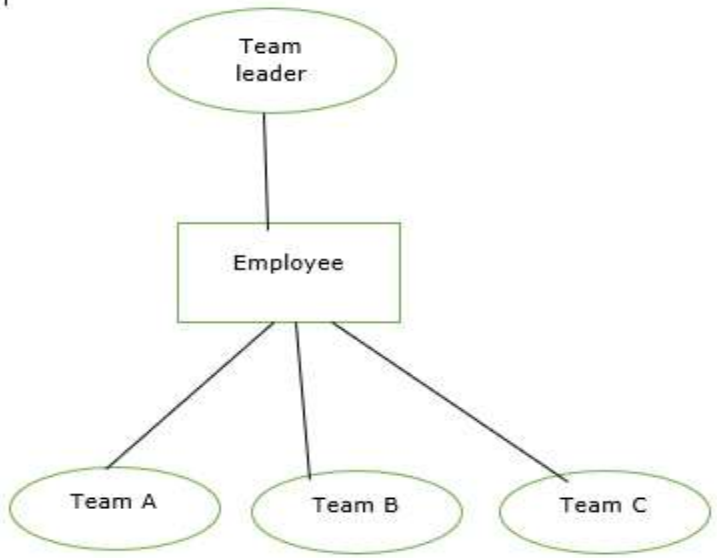
- 2.Attribute defined
- This refers to specifying conditions on more than one attribute.
- Example
- Consider a database for the marks and attendance.
- This type of constraint is defined on two or more attributes for the given entity which is further divided into subclass entities.

Example:



constraints on the generalization and specialization

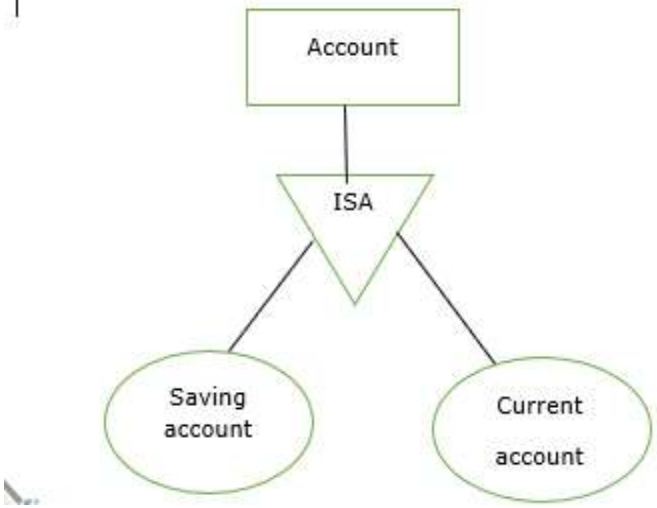
- 3.User defined
- In this constraint the decision is left to the super class that how many instances of the super class will be participating in the sub class.
- In the above example Team leader decides which team will join according to skills.
-



constraints on the generalization and specialization

- 4.Disjoint constraints
- Disjoint is nothing but intersection, the number of instances specified for the given superclass can participate in only one of the sub classes.

Example:



constraints on the generalization and specialization

- 5.Overlapping constraint
- Two or more instances of the super class are participating in two or more sub classes then it is called overlapping constraints.
- Example
- A person who knows Java and PHP can participate in both teams.

constraints on the generalization and specialization

- 6.Complete constraint
- Every instance participates in a relationship.
- All the instances of the superclass must participate in a relationship or into the sub class.

EER

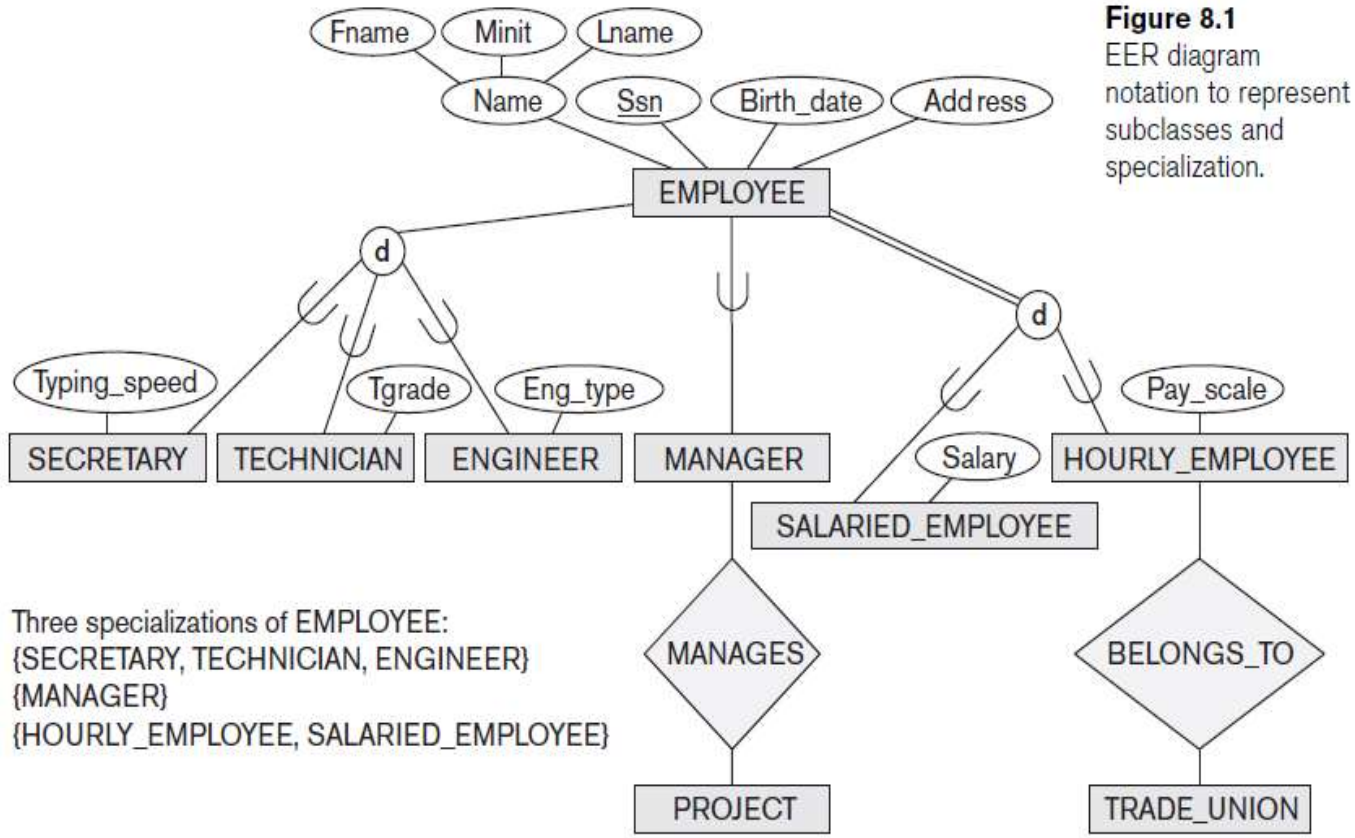


Figure 8.1
EER diagram notation to represent subclasses and specialization.

EER

EER

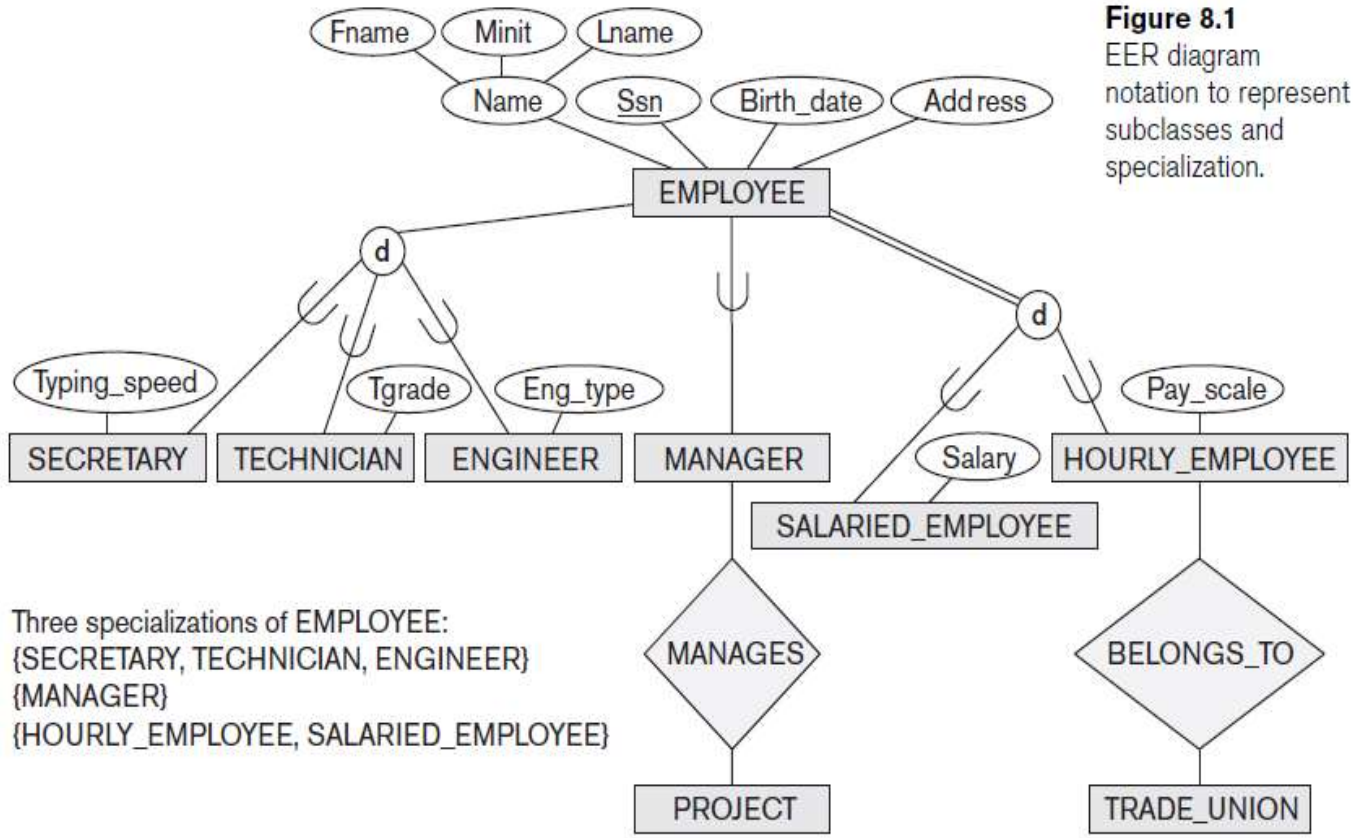


Figure 8.1
EER diagram notation to represent subclasses and specialization.