Database Management System
BIM SEM 4
Unit 2: Entity-Relationship Model (LH 8)

Syllabus: -

Unit 2: Entity-Relationship Model (LH 8)

Entities and Entity Sets. Relationships and Relationship Sets. Attributes. Mapping Constraints. Keys (Super key, Candidate key and Primary key): Primary Keys for Entity Sets and Relationship Sets. The Entity Relationship Diagram. Reducing E-R Diagrams to Tables: Representation of Strong Entity Sets, Representation of Weak Entity Sets, Representation of Relationship Sets. Generalization and Specialization. Aggregation. Mapping Cardinalities: Representation of Mapping Cardinalities in E-R Diagram. Use of Entity or Relationship Sets. Use of Extended E-R Features. Design of an E-R Database Scheme (Case study).

Introduction: The E-R Model

The Entity Relationship data model is based on the perception of the real world that consist of a collection of basics objects called entities, and relationships be tween them. While formulating real-world scenario into the database model, the ER Model creates entity set, relationship set, general attributes and constraints.

The Entity Relationship data model is high-level data model. And was developed by Chen in 1976. It is very simple and easy to design logical view of data. The developer can easily understand the system by looking at an ER model constructed.

ER Model is based on −

- Entities and their attributes.
- Relationships among entities.

Entities and Entity Sets

Entity

An entity is a “thing” or “object” in the real world that is distinguishable from other objects. It can be any object, place, person or activity about which data is recorded. In ER diagram the entity set is represented by rectangle symbol and it is analogues to table in RDBMS.Entities are described in a database by a set of attributes.

Example 1:

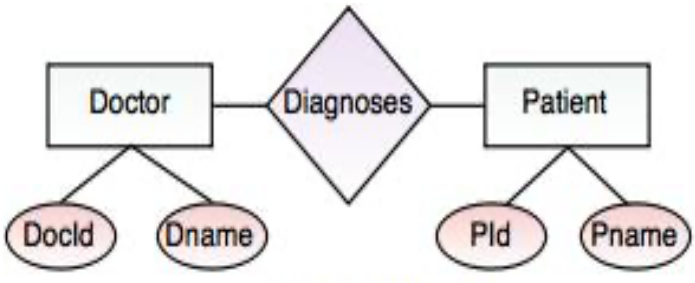


Fig. ER Model

Database Management System
BIM SEM 4

In this diagram, Rectangle represents the entities. Doctor and Patient are considered as an entity. Ellipse represents the attributes. For example, in above diagram, the entity Doctor have attributes like Docid. Dname and entity patient have attribute Pid ,Pname.

An entity can be of two types:

Tangible Entity: Tangible Entities are those entities which exist in the real world physically. Example: Person, car, etc.

Intangible Entity: Intangible Entities are those entities which exist only logically and have no physical existence. Example: Bank Account, etc.

Example 2:

If we have a table of a Student Roll_no, Student_name, Age, Mobile_no then each student in that table is an entity and can be uniquely identified by their Roll Number i.e Roll_no.

Student

Roll_no	Student_name	Age	Mobile_no
1	Andrew	18	7089117222
2	Angel	19	8709054568
3	Priya	20	9864257315
4	Analisa	21	9847852156

Entity

Student Angel with roll number 2 and age 19 is a particular entity as is distinguishable among all the students in the table. Likewise, all the students in the Student Table can be uniquely identified from other students. So, each student in the table is an entity

Entity Type

The entity type is a collection of the entity having similar attributes. In the above Student table example, we have each row as an entity and they are having common attributes i.e each row has its own value for attributes Roll_no, Age, Student_name and Mobile_no.

So, we can define the above STUDENT table as an entity type because it is a collection of entities having the same attributes. So, an entity type in an ER diagram is defined by a name(here, STUDENT) and a set of attributes(here, Roll_no, Student_name, Age, Mobile_no).

Database Management System
BIM SEM 4

The table below shows how the data of different entities (different students) are stored.

Student → Entity Type

Roll_no	Student_name	Age	Mobile_no
1	Andrew	18	7089117222
2	Angel	19	8709054568
3	Priya	20	9864257315
4	Analisa	21	9847852156

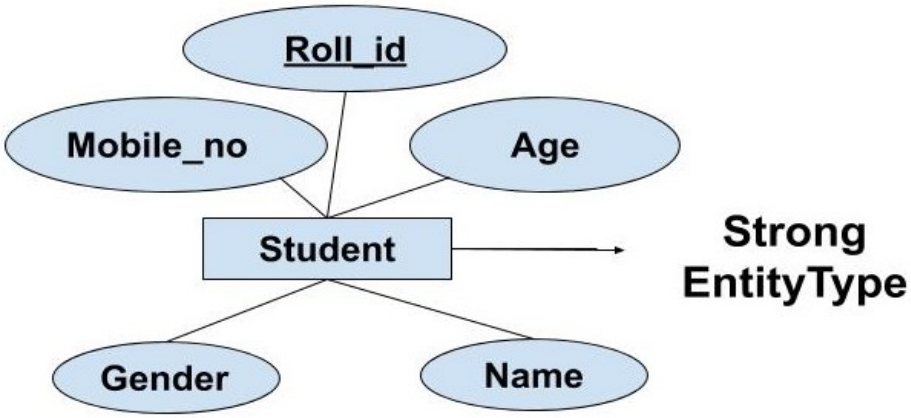
Entity

The E-R representation of the above Student Entity Type is done below.

Types of Entity type

1. **Strong Entity Type**
2. **Weak Entity Type**

Strong Entity Type: Strong entity are those entity types which has a key attribute. The primary key helps in identifying each entity uniquely. It is represented by a rectangle. In the above example, Roll_no identifies each element of the table uniquely and hence, we can say that STUDENT is a strong entity type.

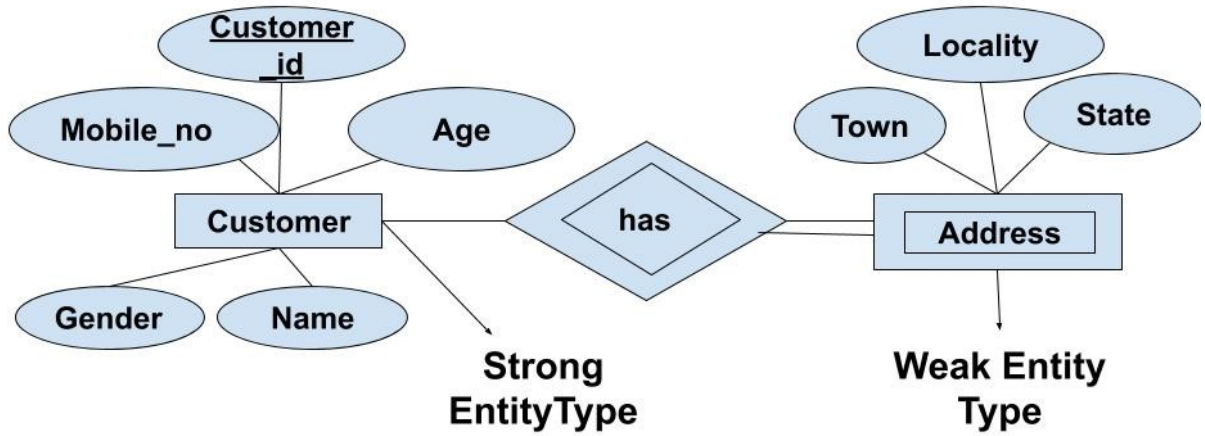


Weak Entity Type: Weak entity type doesn't have a key attribute. Weak entity type can't be identified on its own. It depends upon some other strong entity for its distinct identity.

This can be understood with a real-life example. There can be children only if the parent exits. There can be no independent existence of children. There can be a room only if building exits. There can be no independent existence of a room.

A weak entity is represented by a double outlined rectangle. The relationship between a weak entity type and strong entity type is called an **identifying relationship** and shown with a double outlined diamond instead of a single outlined diamond. This representation can be seen in the diagram below.

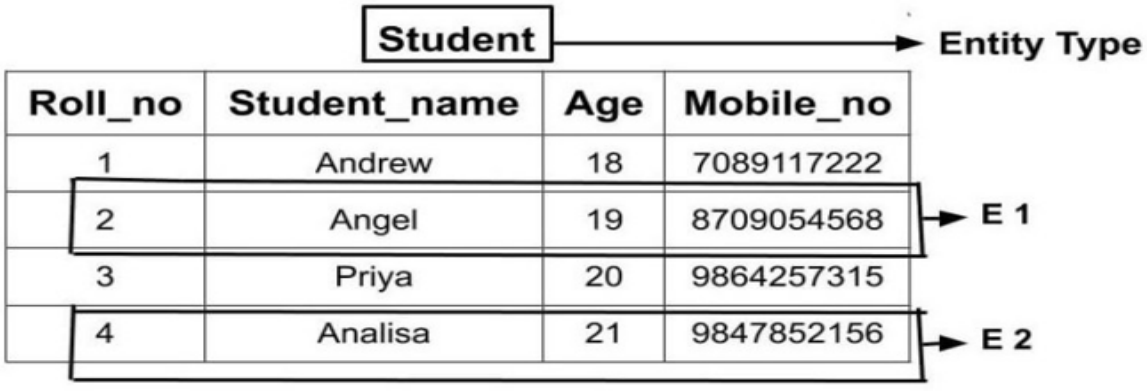
Example: If we have two tables of Customer i.e. Customer(Customer_id, Name, Mobile_no, Age, Gender) and Address(Locality, Town, State, **Customer_id**). Here we cannot identify the address uniquely as there can be many customers from the same locality. So, for this, we need an attribute of Strong Entity Type i.e 'Customer id' here to uniquely identify entities of 'Address' Entity Type.

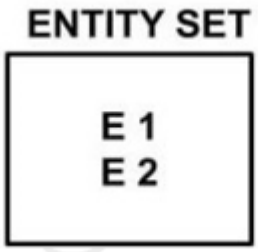


Entity Sets

Entity Set is a collection of entities of the same entity type. Each entity in an entity set has its own set of values for the attributes which make it distinct from other entities in a table. No two entities in an entity set will have the same values for the attributes. In a database, an entity set is represented by the Table.

Example 1: -

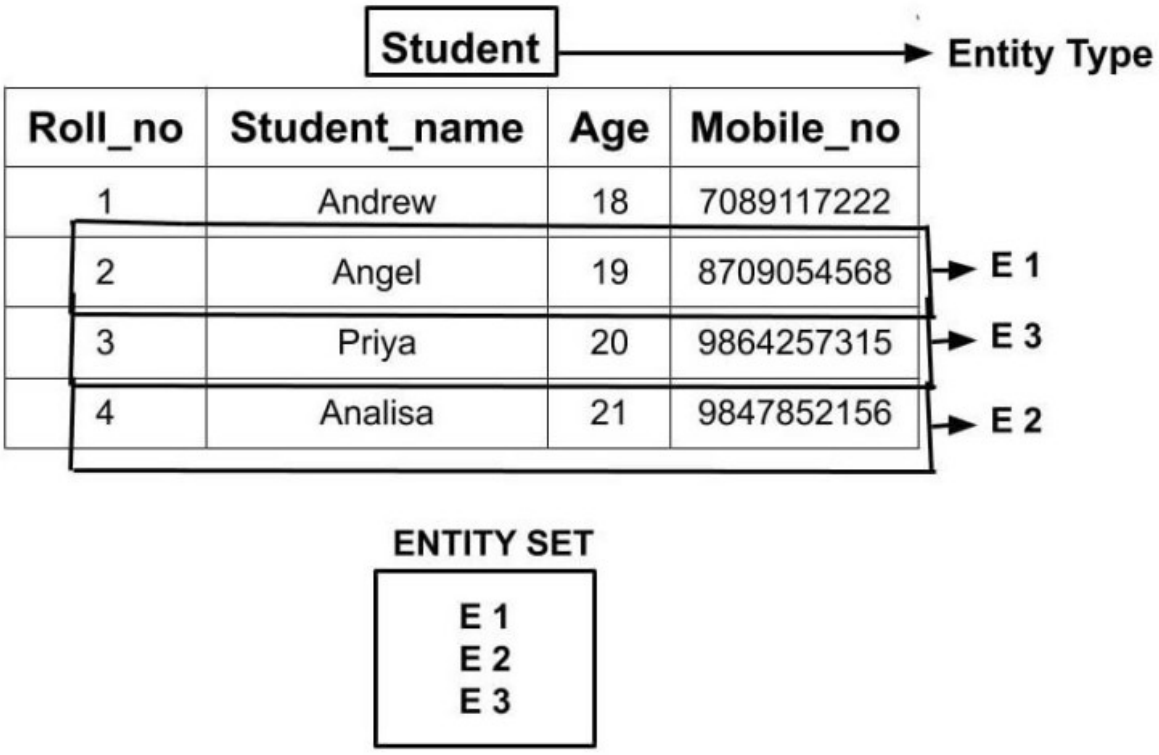




In the above example of STUDENT entity type, a collection of entities from the Student entity type would form an entity set. Here, two entities E1 (2, Angel, 19, 8709054568) and E2(4, Analisa, 21, 9847852156) form an entity set. So, we can say that entity type is a superset of the entity set as all the entities are included in the entity type.

Similar, we can form any combination of the entity set using any of the entities from the entity type 'STUDENT'. Also, we can understand that if we take all the records to the entity set we get the entity type 'STUDENT'. So, we can say that the entity type is the superset of the entity set.

Example 2:



Here, We can form another entity set by taking three entities from the table. (2, Angel, 19, 8709054568), (3, Priya, 20, 9864257315) and (4, Analisa, 21, 9847852156) can also form a entity set.

Database Management System
BIM SEM 4

Relationships and Relationship Sets

Relationships

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

Example: - Relationship between teacher and student

Here teaches is the association between teacher and student. hence **teaches** is called relationships.

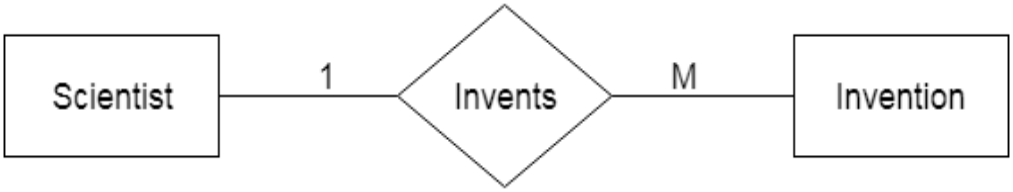
Types of relationship are as follows:

a. One-to-One Relationship

When only one instance of an entity is associated with the relationship, then it is known as one to one relationship. For example, A female can marry to one male, and a male can marry to one female.

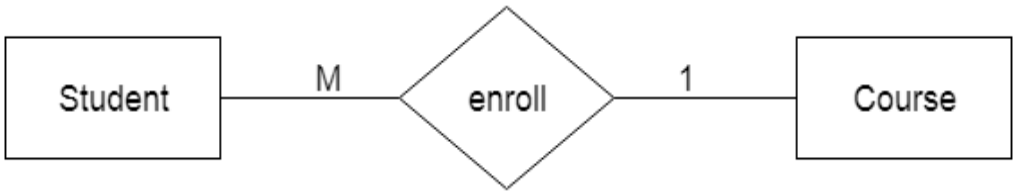
b. One-to-many relationship

When only one instance of the entity on the left, and more than one instance of an entity on the right associates with the relationship then this is known as a one-to-many relationship. For example, Scientist can invent many inventions, but the invention is done by the only specific scientist.



c. Many-to-one relationship

When more than one instance of the entity on the left, and only one instance of an entity on the right associates with the relationship then it is known as a many-to-one relationship. For example, Student enrolls for only one course, but a course can have many students.



d. Many-to-many relationship

When more than one instance of the entity on the left, and more than one instance of an entity on the right associates with the relationship then it is known as a many-to-many relationship. For example, Employee can assign by many projects and project can have many employees.

Relationship Sets

A set of relationship of similar type is called relationship set. Relationship set is the collection of similar relationship. A relationship set can be thought of as a set of n-tuples:

$$\{(e_1, \ldots, e_n) \mid e_1 \in E_1, \ldots, e_n \in E_n\}$$

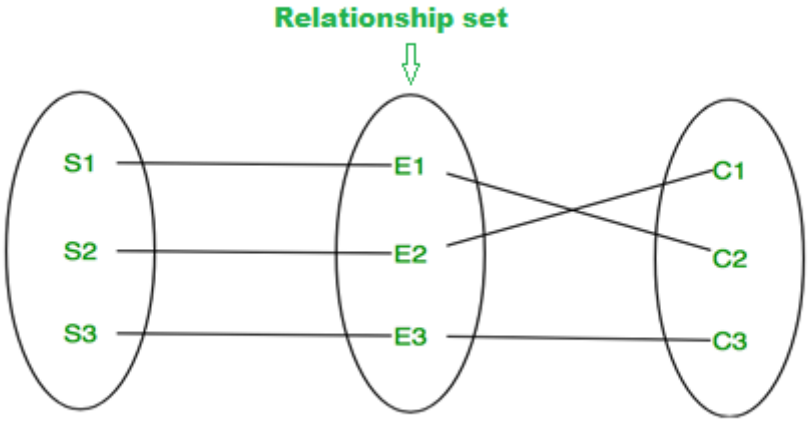
Each n-tuple denotes a relationship involving n entities e1 through en, were entity ei is in entity set Ei .

Example: -

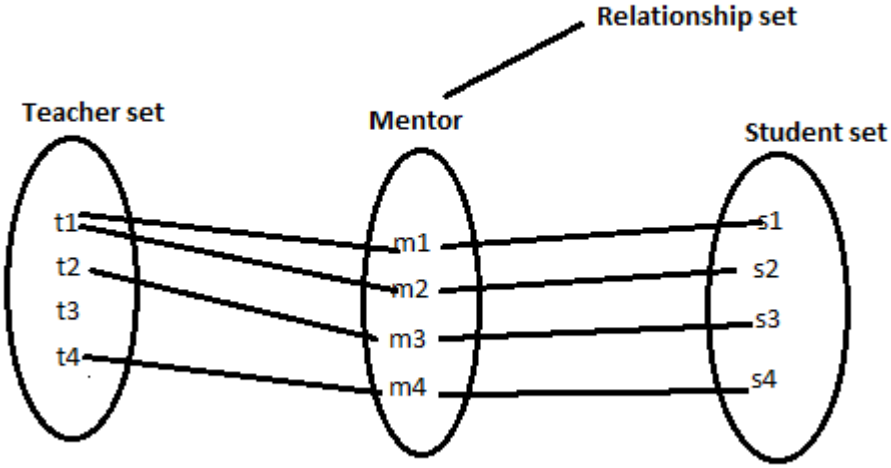
For example, 'Enrolled in' is a relationship type that exists between entity type Student and Course. In ER diagram, relationship type is represented by a diamond and connecting the entities with lines.

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

Database Management System
BIM SEM 4



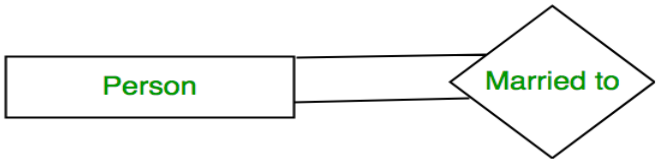
Another Example:



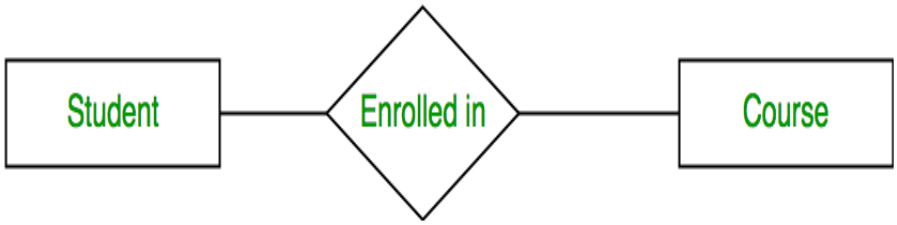
Degree of a relationship set:

The number of different entities sets participating in a relationship set is called as degree of a relationship set.

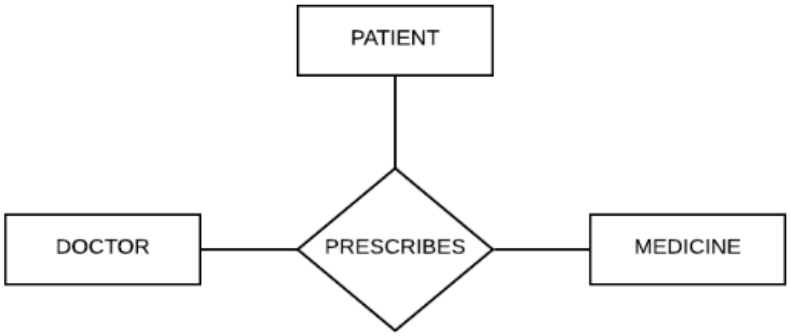
1. Unary Relationship –
   When there is only ONE entity set participating in a relation, the relationship is called as unary relationship. For example, one person is married to only one person.



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



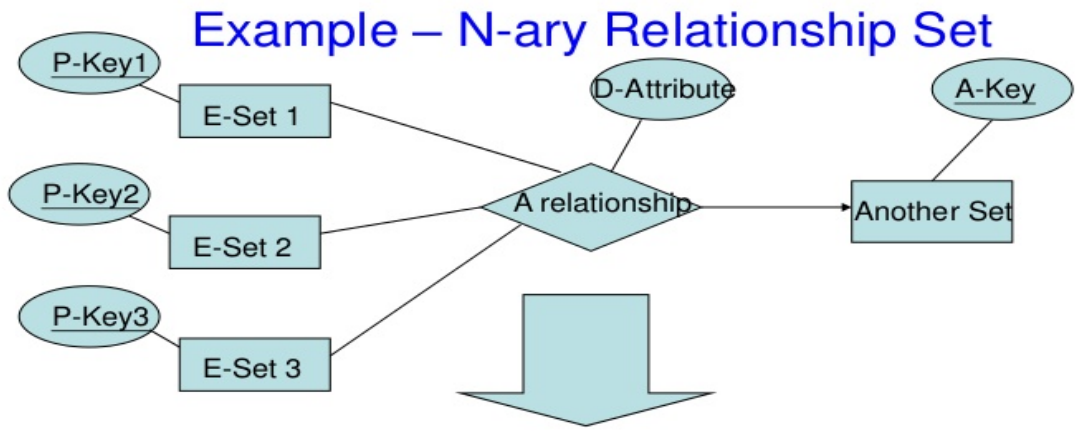
3. Ternary Relationship
   When there is a relationship between three different entities, it is known as a ternary relationship. An example of a ternary relationship can be shown as follows −



In this example, there is a ternary relationship between Doctor, Patient and Medicine.

4. n-ary Relationship –
   When there are n entities set participating in a relation, the relationship is called as n-ary relationship.

Database Management System
BIM SEM 4



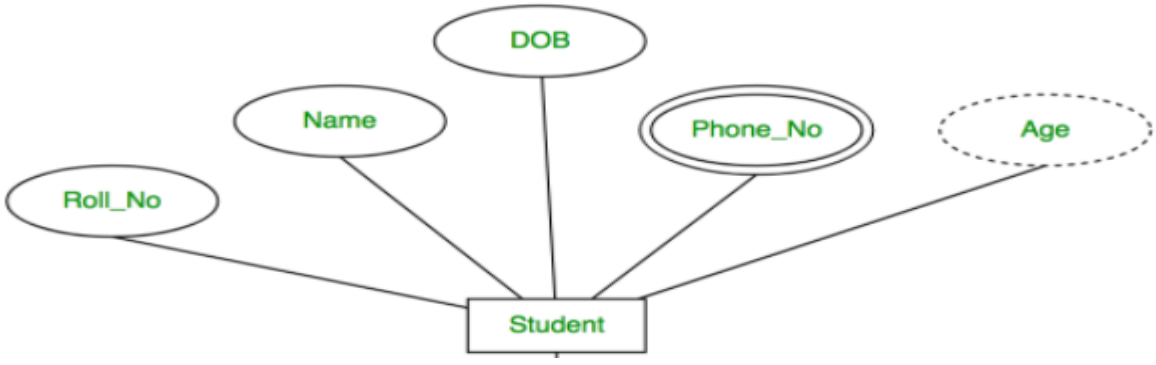
P-Key1	P-Key2	P-Key3	A-Key	D-Attribute
9999	8888	7777	6666	Yes
1234	5678	9012	3456	No

* Primary key of this table is *P-Key1 + P-Key2 + P-Key3*

Attributes: -

An attribute describes the property of an entity. An attribute is represented as Oval in an ER diagram.

For example, Roll_No, Name, DOB, Age, Phone_No are the attributes which defines entity type Student.



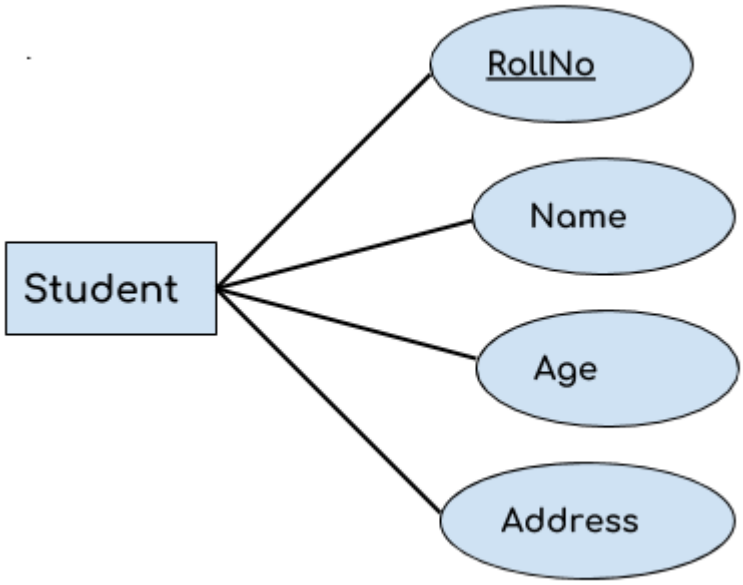
There are four types of attributes:

1. Key attribute
2. Composite attribute
3. Multivalued attribute
4. Derived attribute

Database Management System
BIM SEM 4

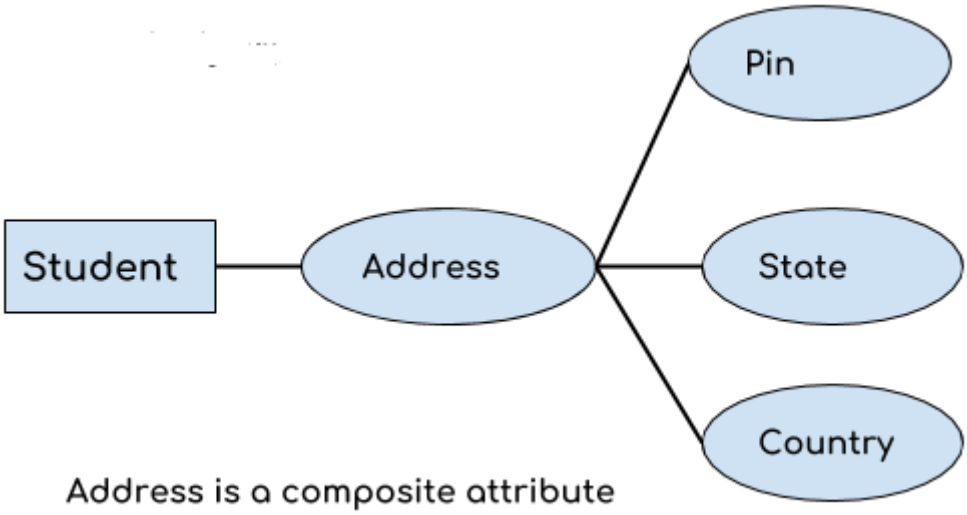
1. Key attribute:

A key attribute can uniquely identify an entity from an entity set. For example, student roll number can uniquely identify a student from a set of students. Key attribute is represented by oval same as other attributes however the text of key attribute is underlined.



2. Composite attribute:

An attribute that is a combination of other attributes is known as composite attribute. For example, In student entity, the student address is a composite attribute as an address is composed of other attributes such as pin code, state, country.



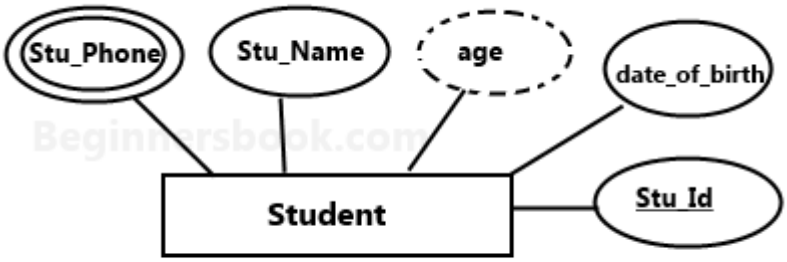
3. Multivalued attribute:

An attribute that can hold multiple values is known as multivalued attribute. It is represented with double ovals in an ER Diagram. For example – A person can have more than one phone numbers so the phone number attribute is multivalued.

Database Management System
BIM SEM 4

4. Derived attribute:

A derived attribute is one whose value is dynamic and derived from another attribute. It is represented by dashed oval in an ER Diagram. For example – Person age is a derived attribute as it changes over time and can be derived from another attribute (Date of birth).E-R diagram with multivalued and derived attributes:



Mapping Constraints

Mapping constraints defines how many entities can be related to another entity t a relationship.

Or

Mapping constraint of relationship is the number of possible occurrences of any entity set that may relate to a single or multiple entity sets with relationship. It is very much useful for identifying relationships that are involved with more than one relationship

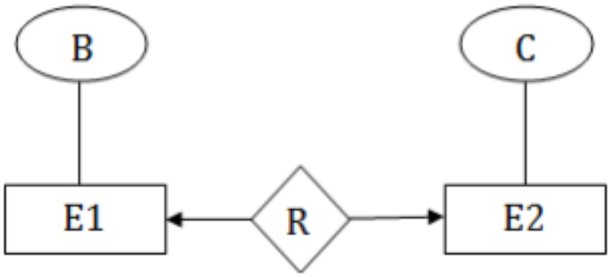
Hence,

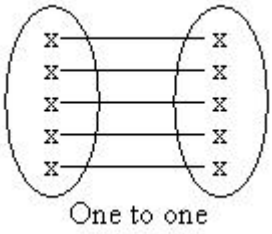
- A mapping constraint is a data constraint that expresses the number of entities to which another entity can be related via a relationship set.
- It is most useful in describing the relationship sets that involve more than two entity sets.
- Mapping constraints can be explained in terms of mapping cardinality. For binary relationship set R on an entity set A and B, there are four possible mapping cardinalities. These are as follows:
    1. One to one (1:1)
    2. One to many (1:M)
    3. Many to one (M:1)
    4. Many to many (M:M)

Database Management System
BIM SEM 4

One-to-one

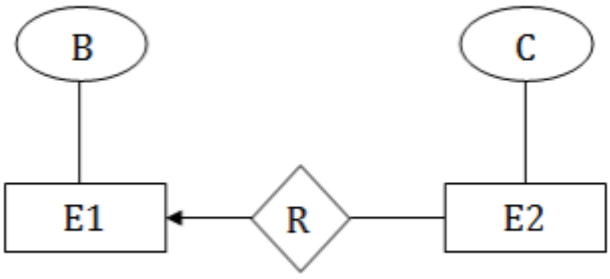
- When a single instance of an entity is associated with a single instance of another entity, then it is called as one to one cardinality
- Here each entity of the entity set participate only once in the relationship
- eg. In one-to-one mapping, an entity in E1 is associated with at most one entity in E2, and an entity in E2 is associated with at most one entity in E1.



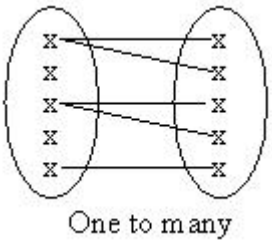


One-to-many

- When a single instance of an entity is associated with more than one instance of another entity then this type of relationship is called one to many relationships.
- Here entities in one entity set can take participation in any number of times in relationships set and entities in another entity set can take participation only once in a relationship set
- Eg.In one-to-many mapping, an entity in E1 is associated with any number of entities in E2, and an entity in E2 is associated with at most one entity in E1

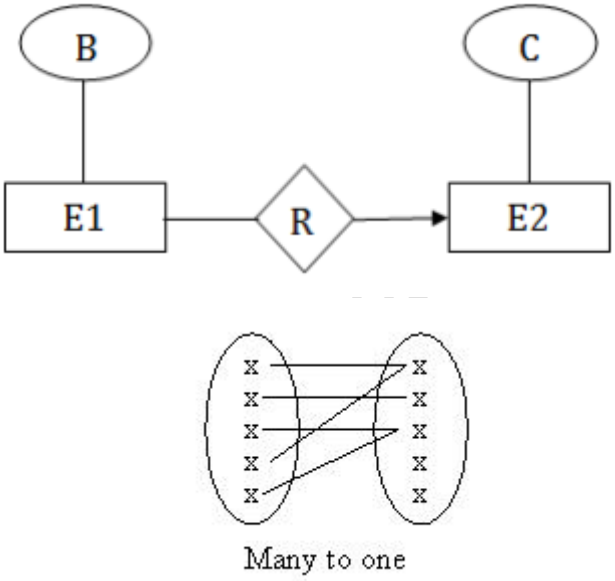


Database Management System
BIM SEM 4



Many-to-one

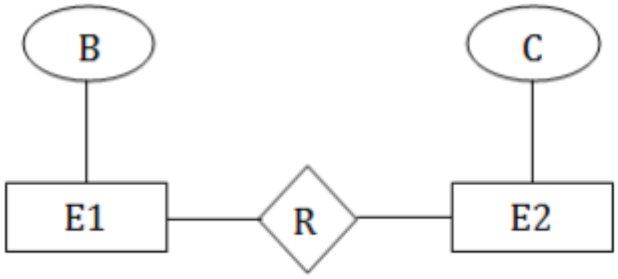
- When entities in one entity set can participate only once in a relationship set and entities in another entity can participate more than once in the relationship set, then such type of cardinality is called many-to-one
- Eg.In one-to-many mapping, an entity in E1 is associated with at most one entity in E2, and an entity in E2 is associated with any number of entities in E1.

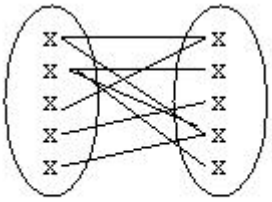


Many-to-many

- Here more than one instance of an entity is associated with more than one instance of another entity then it is called many to many relationships
- In this cardinality, entities in all entity sets can take participate any number of times in the relationship cardinality is many to many.
- Eg.In many-to-many mapping, an entity in E1 is associated with any number of entities in E2, and an entity in E2 is associated with any number of entities in E1.

Database Management System
BIM SEM 4

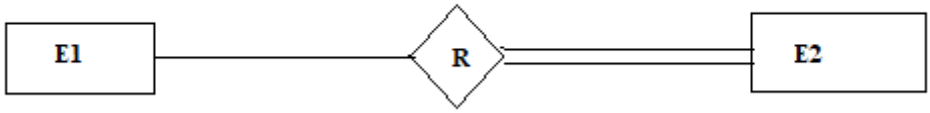




Many to many

Participation Constraints:

1. The participation of an entity set E in a relationship set R is said to be ' total' if every entity in E participate in at least one relationship in R.
2. If only some entities in E participate in relationship in R, then entity set E in relationship R is said to be partial.



(I) Total participation of E2 in R

(II) Partial Participation of E1 in R

Note: Double line indicates total participation of entity set in a relationship set.

Example:

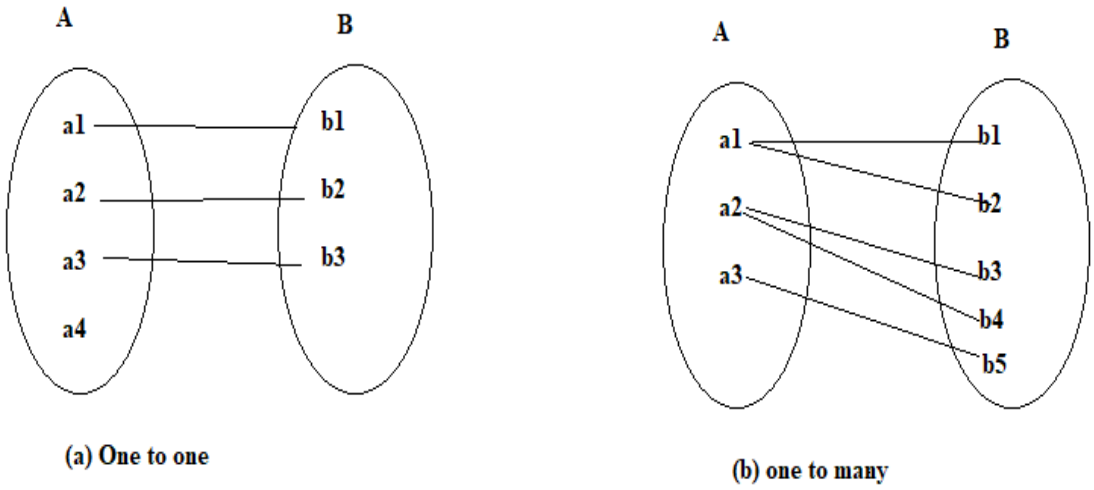


Figure: mapping Cardinalities

In above figure (a) the participation of B in relationship set is **total** while the participation of A in relationship set is partial. And similarly in figure(b) the participation of both A and B in relationship set is total. So above figure can be represent as mapping constraint as….

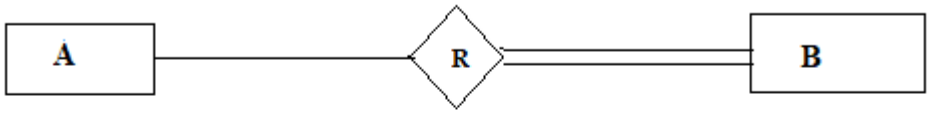


Figure (a): particial participation of A in R

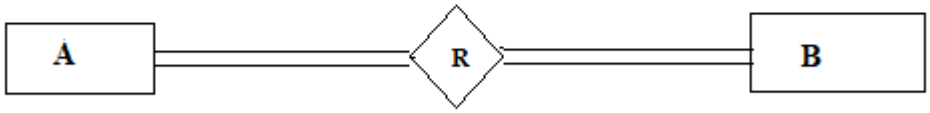


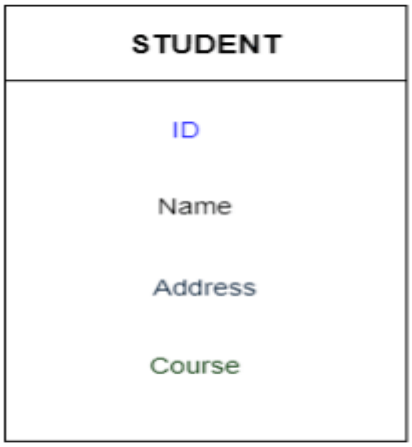
Figure (b): Total participation of A and B in R

Keys (Super key, Candidate key and Primary key):

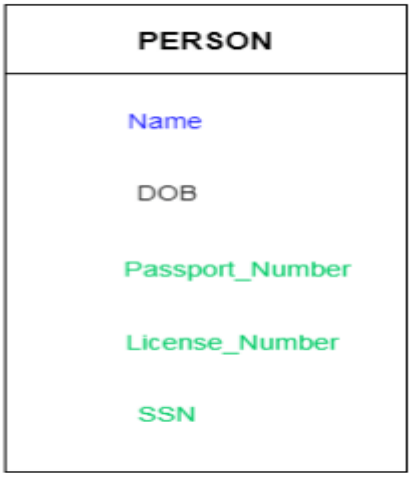
Keys

- Keys play an important role in the relational database.
- It is used to uniquely identify any record or row of data from the table. It is also used to establish and identify relationships between tables.

- For example: In Student table, ID is used as a key because it is unique for each student.

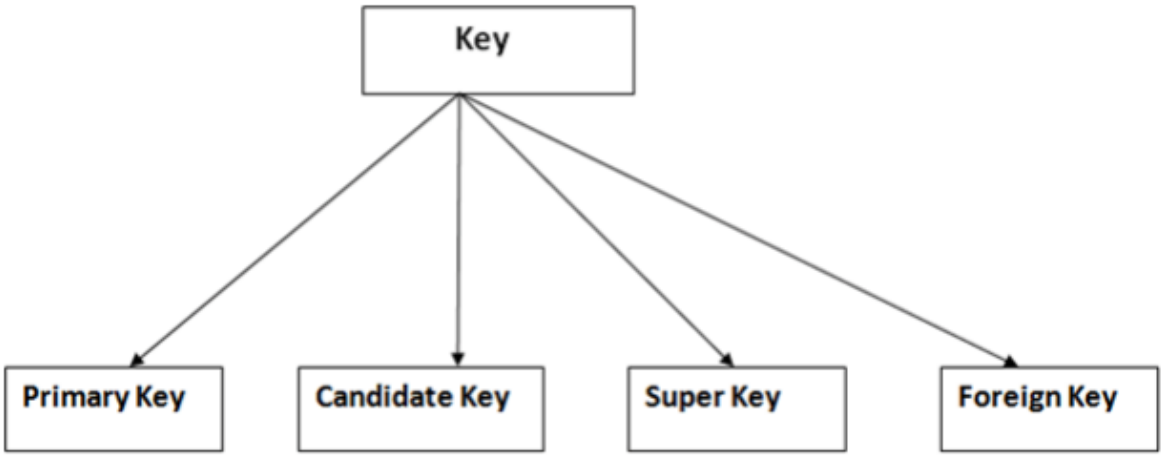


- Similarly, in PERSON table, passport_number, license_number, SSN are keys since they are unique for each person.



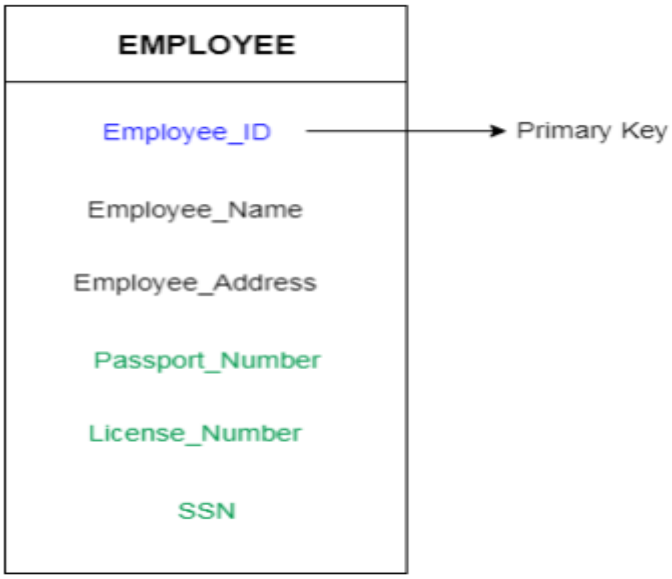
Types of key:

Database Management System
BIM SEM 4



1.Primary key

- A primary key uniquely identifies each record in a table and must never be the same for two or more records.
- It is the first key which is used to identify one and only one instance of an entity uniquely.
- Primary key does not permit Null value.
- In the EMPLOYEE table, ID can be primary key since it is unique for each employee.
- In the EMPLOYEE table, we can even select License_Number and Passport_Number as primary key since they are also unique.
- For each entity, selection of the primary key is based on requirement and developers.
- Consider the following table.

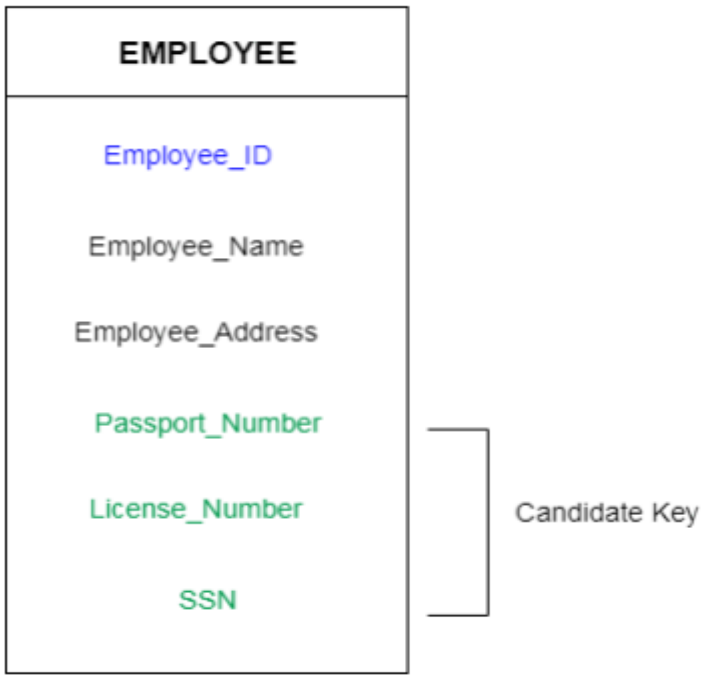


Here, Employee_ID is the primary key

2.Candidate key

smn

- Any attribute (set of attributes) that uniquely identifies a row in a table is a candidate key for the primary key then such an attribute is called candidate key.
- OR, A candidate key is an attribute or set of an attribute which can uniquely identify a tuple.
- The remaining attributes except for primary key are considered as a candidate key. The candidate keys are as strong as the primary key.
- For example: In the EMPLOYEE table, Employee_ID is best suited for the primary key. Rest of the attributes like SSN, Passport_Number, and License_Number, etc. are considered as a candidate key.



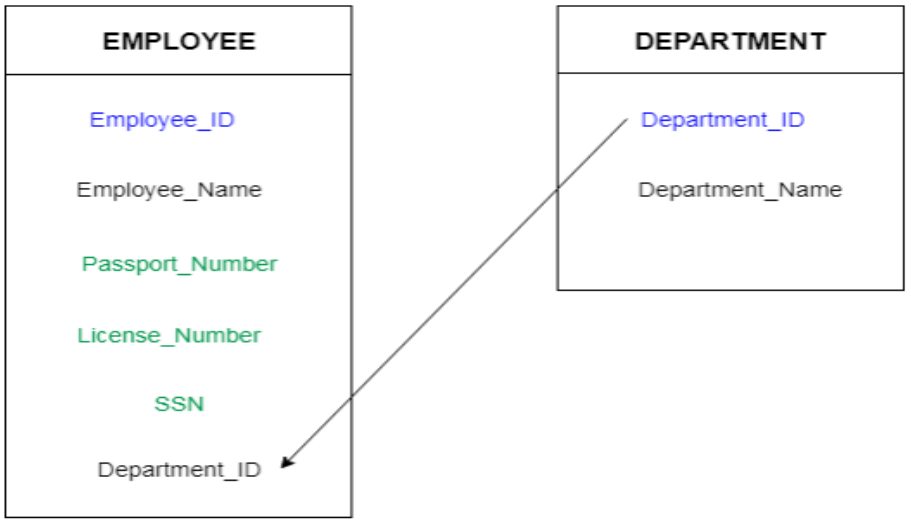
3.Super Key

- Super key is a set of one or more than one keys that can be used to uniquely identify the record in a table.
- Or, A Super key is a set of an attribute which can uniquely identify a tuple.
- Super key is a superset of a candidate key.
- A super key for an entity is a set of one or more attribute whose combined value uniquely identifies the entity in the entity set.
- A super key can be combined form of primary key, alternate key, unique key etc.
- For example: In the above EMPLOYEE table, for (EMPLOEE_ID, EMPLOYEE_NAME) the name of two employees can be the same, but their EMPLYEE_ID can't be the same. Hence, this combination can also be a key. The super key would be EMPLOYEE-ID, (EMPLOYEE_ID, EMPLOYEE-NAME), etc.

4.Foreign key

- A FOREIGN KEY is a key used to link two tables together.
- A FOREIGN KEY is a field (column) in one table that refers to the PRIMARY KEY in another table.

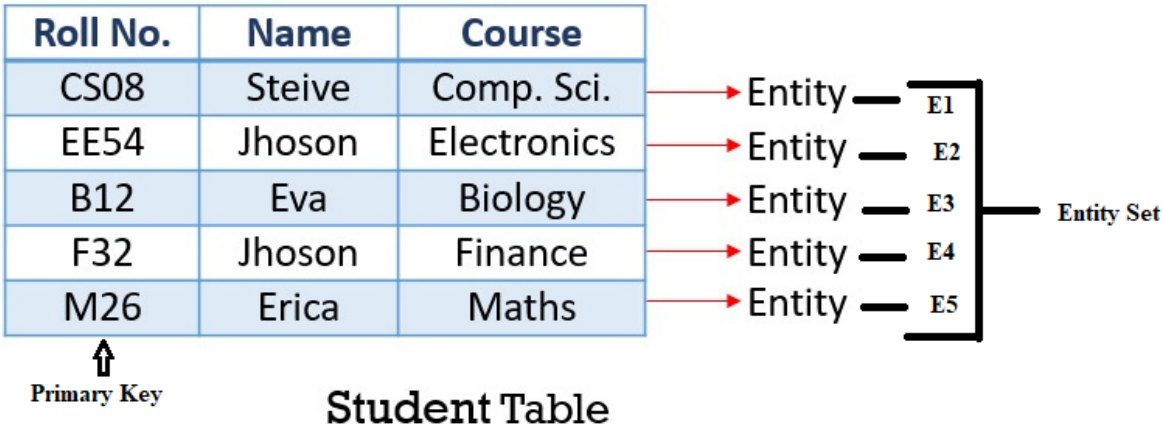
- The table containing the foreign key is called the child table, and the table containing the candidate key is called the referenced or parent table.
- In a company, every employee works in a specifc department, and employee and department are two diferent entities. So we can't store the information of the department in the employee table. That's why we link these two tables through the primary key of one table.
- We add the primary key of the DEPARTMENT table, Department_Id as a new attribute in the EMPLOYEE table.
- Now in the EMPLOYEE table, Department_Id is the foreign key, and both the tables are related.



Primary Keys for Entity Sets and Relationship Sets

Primary Keys for Entity Sets

consider the relation (or entity type) student shown in figure below.



smn

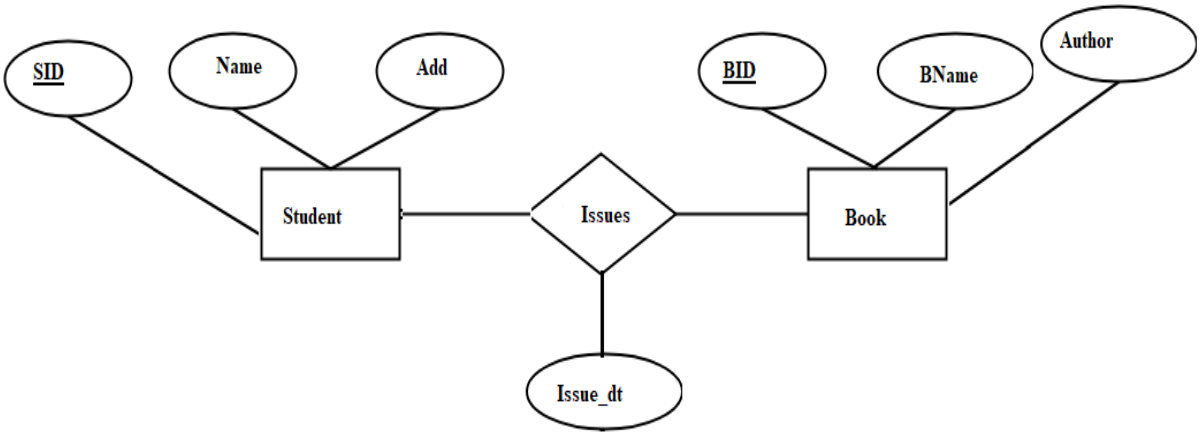
Now, as we know, an entity set is a group of entities that possess the same set of attributes. Each entity in an entity set has its own set of values for the attributes which make it distinct from other entities in a table. No two entities in an entity set will have the same values for the attributes.

In a database, an entity set is represented by the Table. Above you can see the Student Table which have multiple entries i.e. entity or all the entities can be considered as entity set.

in ER diagram an entity set is always represented with the rectangle.

As we know a primary key uniquely identifies each record in a table and must never be the same for two or more records. Now, observe that in above student table each student has different roll number and it uniquely identifies each entity or set of entity. hence, here roll no is a primary key for entity set.

Primary Keys for Relationship Sets



The primary key in a relationship set is a combination of the primary keys of all entity sets participating in the relationship set. The attributes of a relationship set are the attributes that comprise the primary keys of the entity sets involved in the relationship set.

For example:

- SID is the primary key of student, and
- BID is the primary key of Book.
- The attributes of the relationship set Issues are then (SID, BID.).

This is enough information to enable to relate a student to a book. If the relationship has descriptive attributes, those are also included in its attribute set.

For example, we might add the attribute **Issue_dt** to the above relationship set, signifying the date of issue a book by a particular student. Note that this attribute cannot instead be placed in either entity set as it relates to both a student and a book, and the relationship is many-to-many.

Database Management System
BIM SEM 4

The primary key of a relationship set R depends on the mapping cardinality and the presence of descriptive attributes.

If there are no any descriptive attributes in the relationship then the primary keys will be….

- **In many-to-many relationship:** all attributes in R.
- **In one-to-many relationship:** primary key for the ``many'' entity.

Descriptive attributes may be added, depending on the mapping cardinality and the semantics involved. So, if above ER diagram can be decomposed, then, the following relation will be made.

Student

SID	Name	Add

Book

BID	BName	Author

Issues

SID	BID	Issue_dt

Here, primary key will be considered as combination of SID and BID.

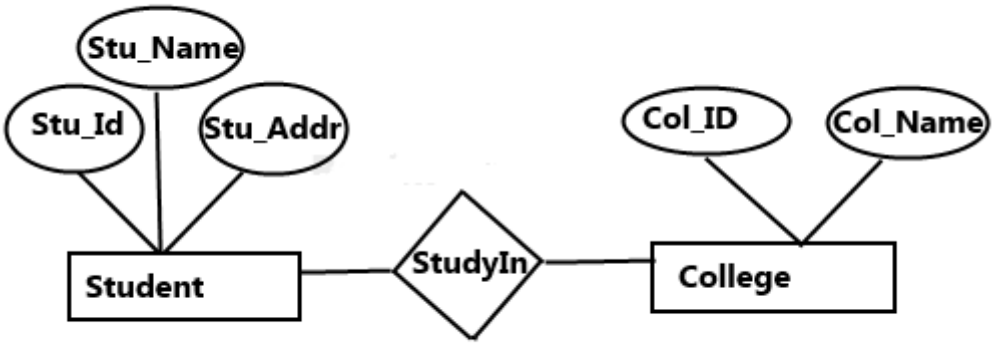
The Entity Relationship Diagram

Entity relationship diagram also known as ER diagram. It is a type of structural diagram used in database design. ER Diagram displays the relationship of entity sets stored in a database. In other words, ER diagrams help to explain the logical structure of databases. ER diagrams are created based on three basic concepts: entities, attributes and relationships.

An entity set is a group of similar entities and these entities can have attributes. In terms of DBMS, an entity is a table or attribute of a table in database, so by showing relationship

among tables and their attributes, ER diagram shows the complete logical structure of a database.

Example



In the above diagram we have two entities Student and College and their relationship. Student entity has attributes such as Stu_Id, Stu_Name & Stu_Addr and College entity has attributes such as Col_ID & Col_Name.
The relationship between Student and College is many to one as a college can have many students however a student cannot study in multiple colleges at the same time.

There are several notations to draw the ER diagrams, among which the chen's model and crow foot model are most popular ones.

The ER diagram contains different symbols and connectors that visualizes the important information like "the major entities within the system scope" and the "entity relationship among these entities".

Symbols used in ER diagram

- **Rectangles:** This Entity Relationship Diagram symbol represents entity types

- **Ellipses :** Symbol represent attributes

smn

- **Diamonds:** This symbol represents relationship types

- **Lines:** It links attributes to entity types and entity types with other relationship types
- **Primary key:** attributes are underlined

- **Double Ellipses:** Represent multi-valued attributes

Derived Attribute for any Entity

Derived attributes are those which are derived based on other attributes, for example, age can be derived from date of birth.

To represent a derived attribute, another dotted ellipse is created inside the main ellipse.

Weak Entity

A weak Entity is represented using double rectangular boxes. It is generally connected to another entity.

Database Management System
BIM SEM 4

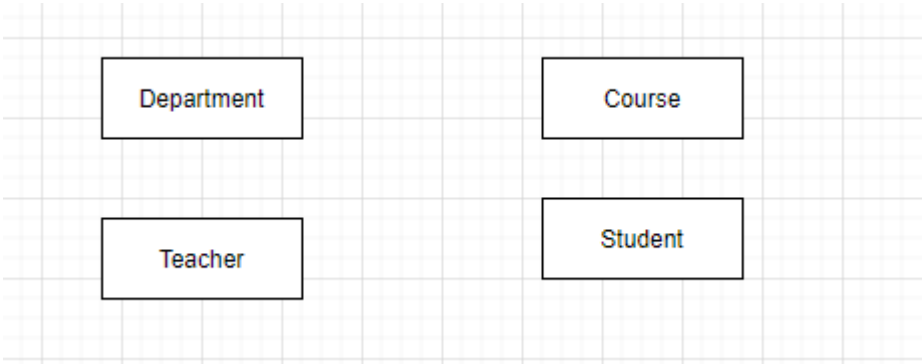
Steps for creating ER diagram

1) Identify the entities
2) Identify the key attributes and other attributes
3) Find the relationship
4) Define the mapping cardinalities
5) Draw the complete ER diagram

Example: (case study)

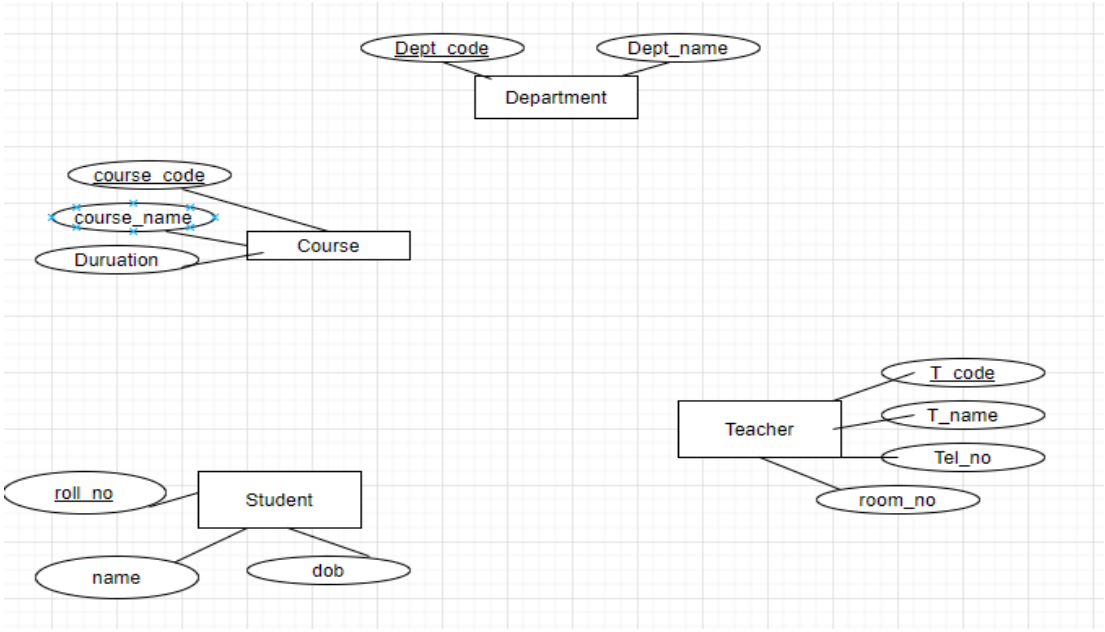
A university has a many **Departments**. Each departments have multiple **teachers** and one among them is head of department. Each department offers multiple **courses**, each being taught by a single teacher. Therefore, one teacher belongs to only one department. Further, A **student** may enroll for many courses offered by different departments. A particular course can be offered by only one department.

1) **Identify the entity**
   Department,Teacher,Course,Student



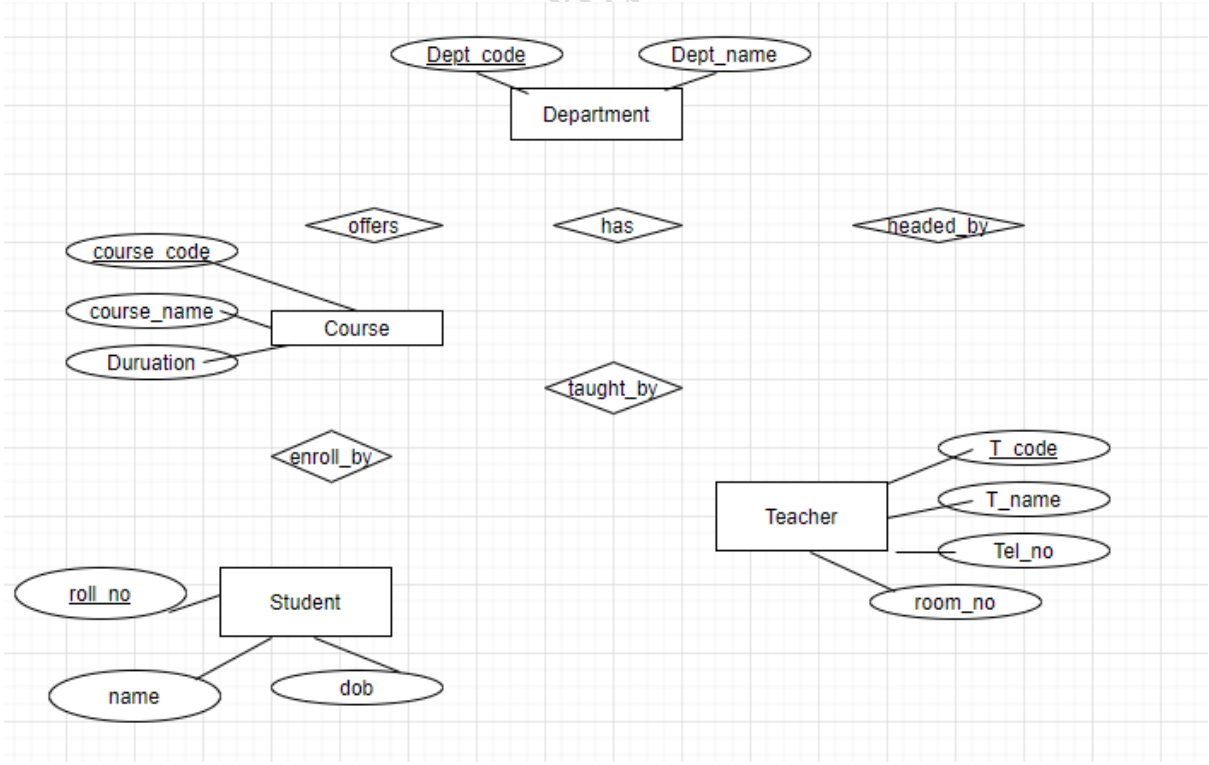
2) **Identify the key attribute and other attributes**
   - Dept_code is a key attribute for Department attribute and other attributes of it are dept_name
   - For teacher, T_code(key attribute) and other are T_name,Tel_no,room_no
   - For course, course_code(key attribute),course_name,duruation
   - For student, roll_no(pk),name,dob

Database Management System
BIM SEM 4



3) Find the relationship

a) Department **offers** multiple course
b) Course **enrolls by** student
c) Department **has** teachers
d) department headed by head of department (i.e teacher)

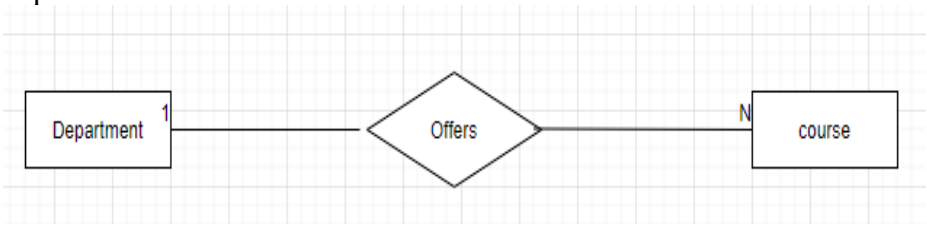


smn

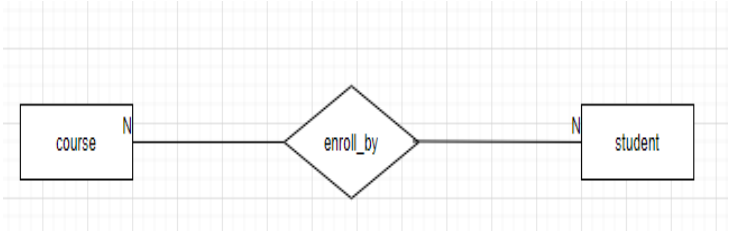
Database Management System
BIM SEM 4

4) define mapping cardinalities

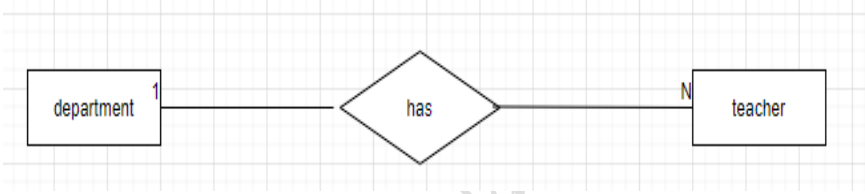
a) The department offers multiple courses and each course belongs to only one department



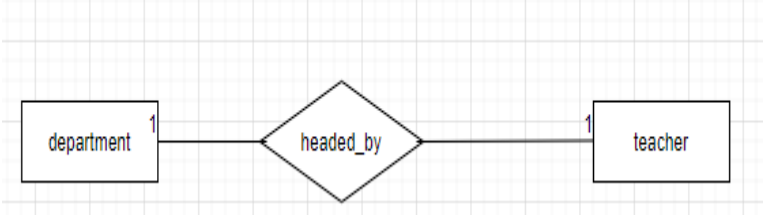
b) multiple student can enroll for multiple course



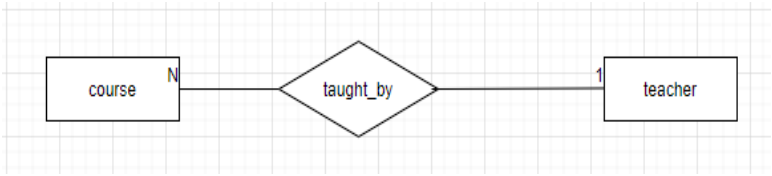
c) one department has multiple teacher



d) each department has one head of department and only one teacher is head of department

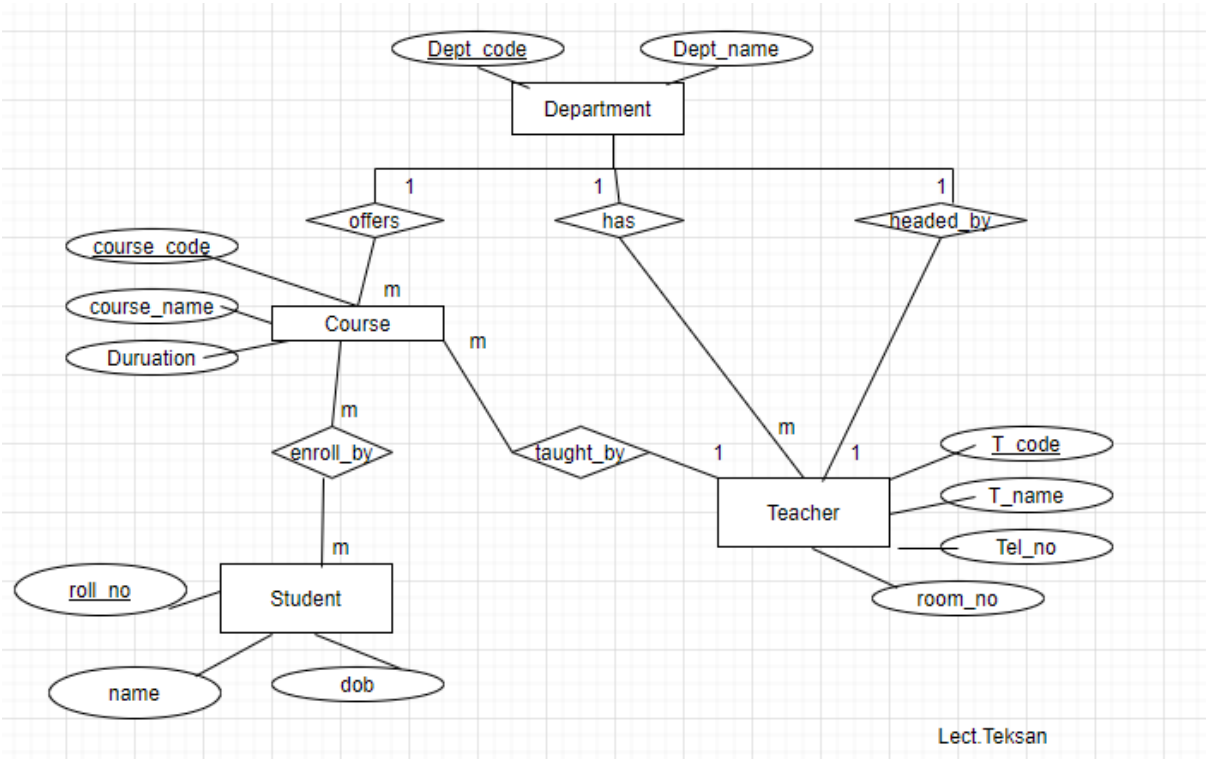


e) one course is taught by only one teacher but teacher teaches many courses



Database Management System
BIM SEM 4

5) Draw the complete Er diagram

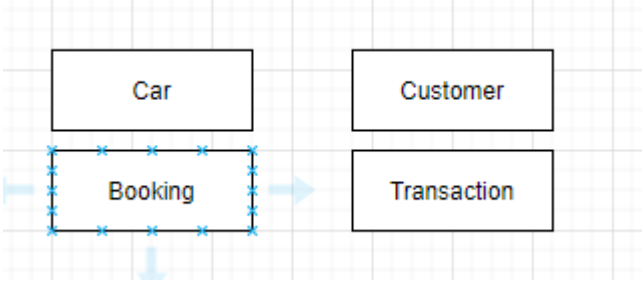


Example: (case study)

"Saboo-Car-Rental-Services" is a **car** rental showroom, who want to automate their business. They offer different types of cars on rent as small car, SUV, MUV. Each type of car has the maximum seating available and the tariff per kilometre. The management wants the system to show availability of the number of cars of each type for serving the inquiry. The system should have a provision for booking the car. Before the booking is made, the **customer** needs to provide personal information and driving license details. **Booking** is typically stored as booking date, date of rent, duration in hours and type of vehicle.

Once the booking is done a unique booking number is provided to the customer for their reference which they need to produce at the time they come to collect the car. A new **transaction** record is created for each booking after the car is returned, specifying the kilometres used and the amount to be paid, date of payment.

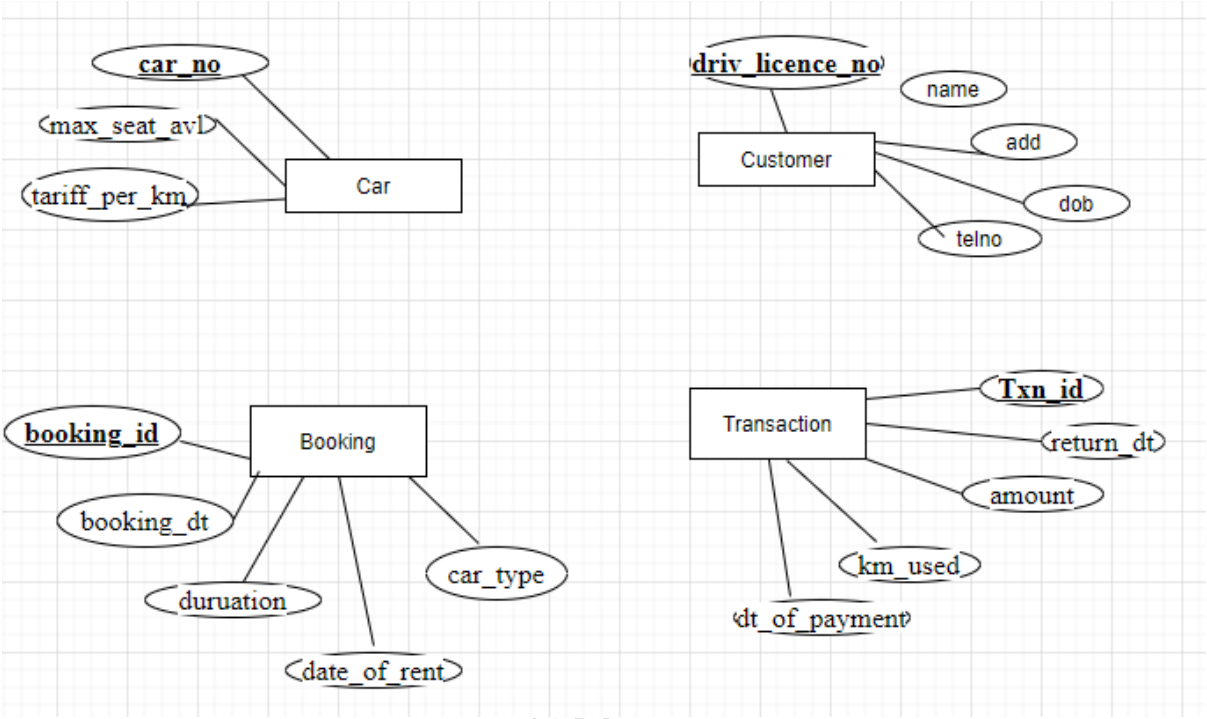
1) **Identify the entity**
   Car,customer,booking and transaction



Database Management System
BIM SEM 4

2) Identify the key attribute and other attributes

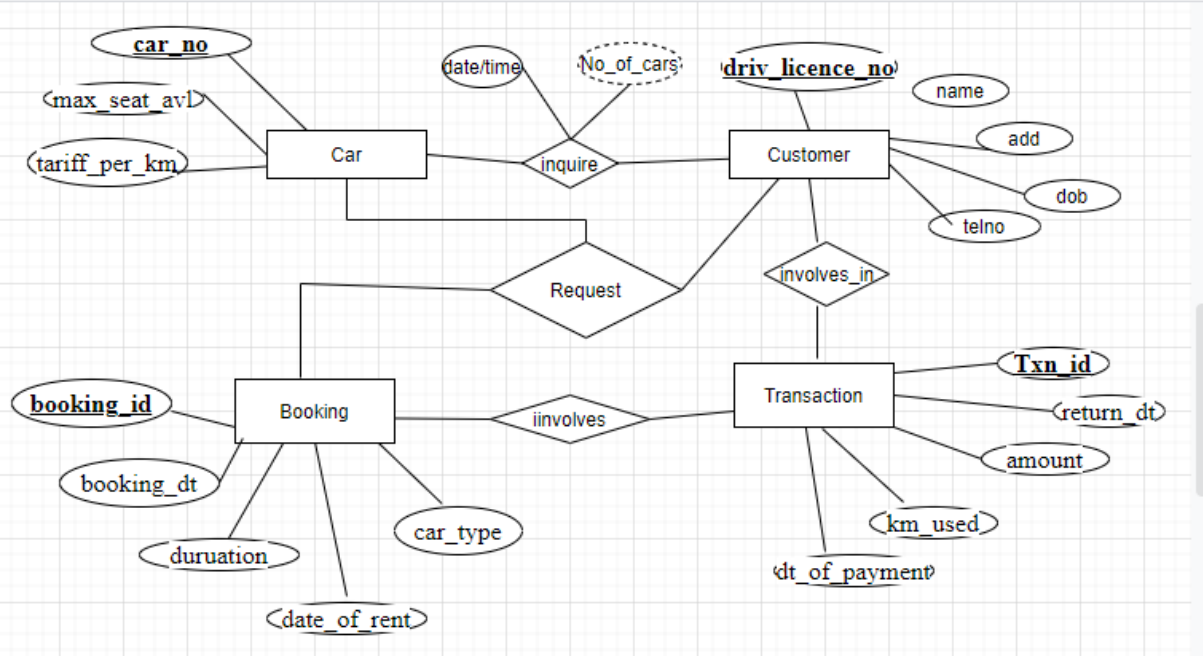
- For car : **car_no**, max_seat_avl, tariff_per_km
- For Customer: name,add,telno,dob,driv_licence_no
- For booking: booking_id, booking_dt, duruation, car_type, date_of_rent
- For transaction: Txn_id, amount, km_used, dt_of_payment, return_dt



3) Find the relationship

a) Customer **inquiries about** the car
b) Customer **request** car for booking
c) Customer **involves** in transaction
d) Booking **leads to/involves** transaction

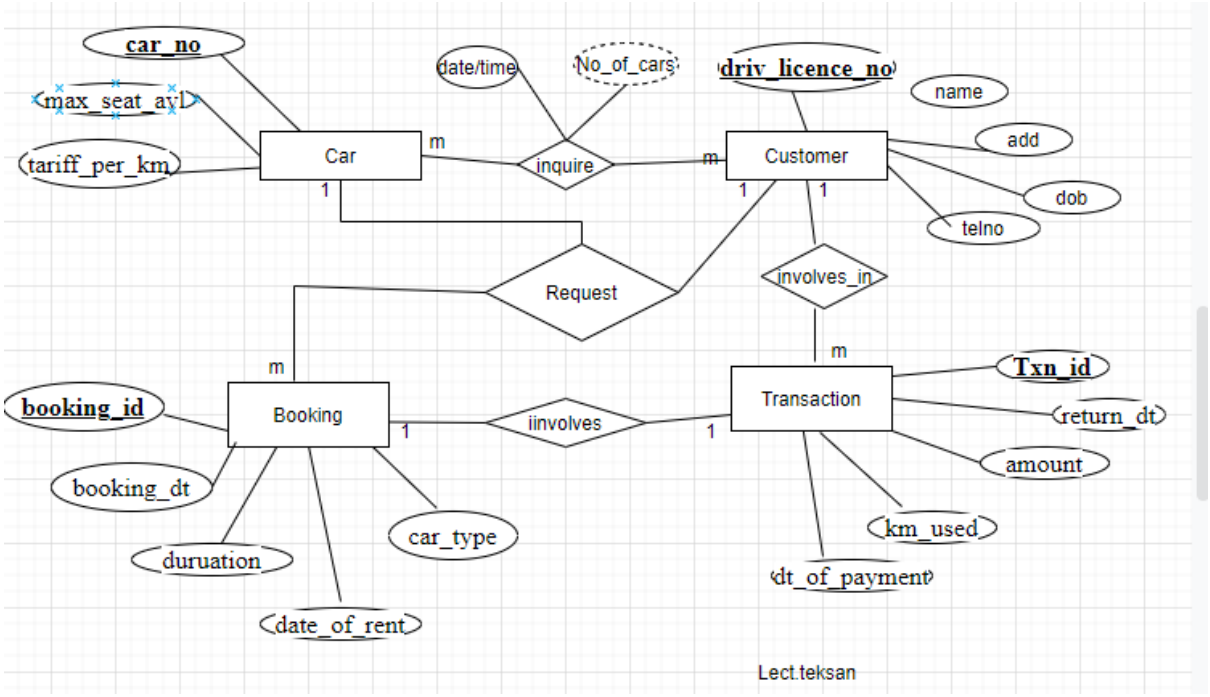
Database Management System
BIM SEM 4



4) Define mapping cardinalities

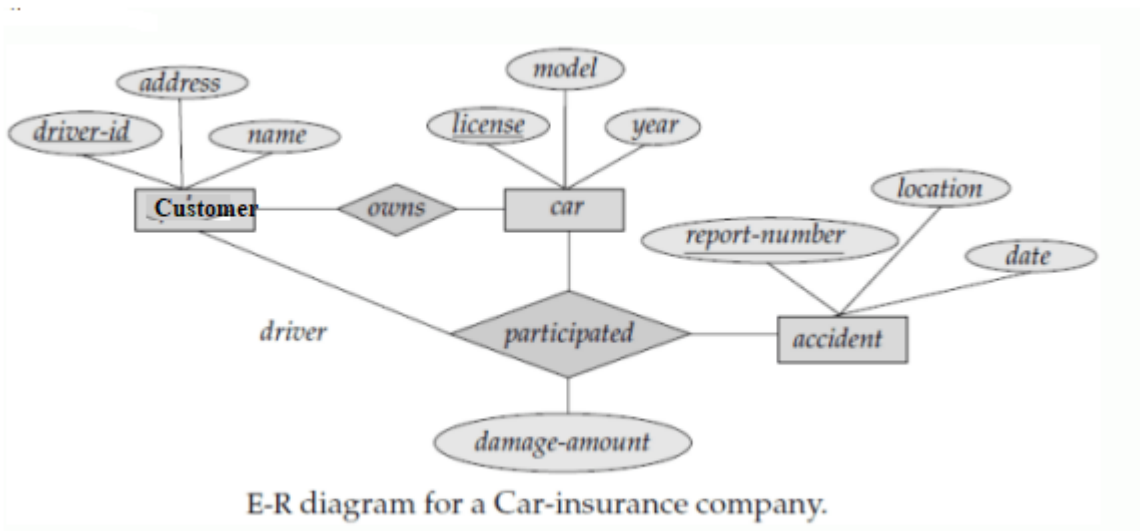
a) One Customer Can involve in many transactions.
b) One Customer Can request for many bookings.
c) One transaction involves one booking.
d) Many Customers Can inquire many cars.
e) One booking can lead to only one transaction. Similarly, one transaction associated with one booking.
f) The availability of a car can be calculated by referring to the booking

Database Management System
BIM SEM 4



Case study 3

Construct an E-R diagram for a car-insurance company whose customers own one or more cars each. Each car has associated with it zero to any number of recorded accidents.
Note: add attribute as your wish/interest



E-R diagram for a Car-insurance company.

smn

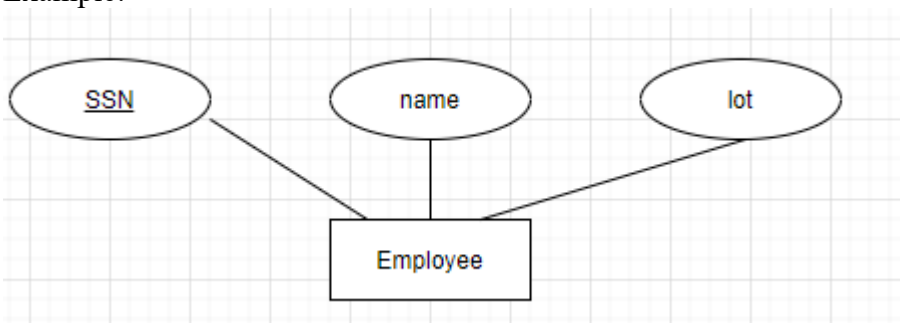
Database Management System
BIM SEM 4

Reducing ER diagrams to table

When the ER diagrams are translated into a set of relations or tables the following thing happens…

1) Entity set:
   Entity set becomes tables
2) Attributes:
   Attributes becomes fields
3) Key Attribute:
   The key attributes correspond to primary key of table.

Example:



Employee

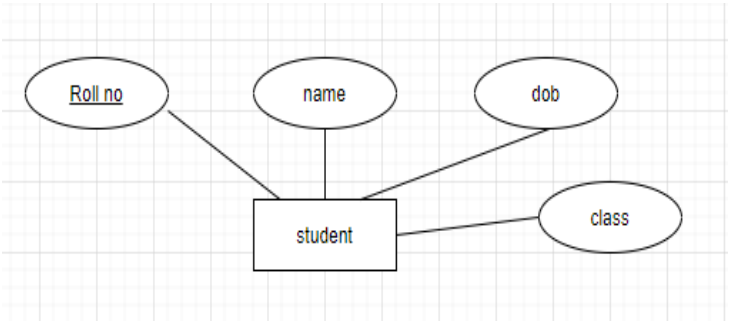
SSN	Name	lot

❖ **Representation of strong entity set**

Strong entity set:

Strong entity set is complete by itself and it is not dependent on any other entity type. It possesses a primary key which describes each instance in the strong entity set uniquely. i.e., strong entity set can be uniquely identified. it is represented by a square with single line.

Example:

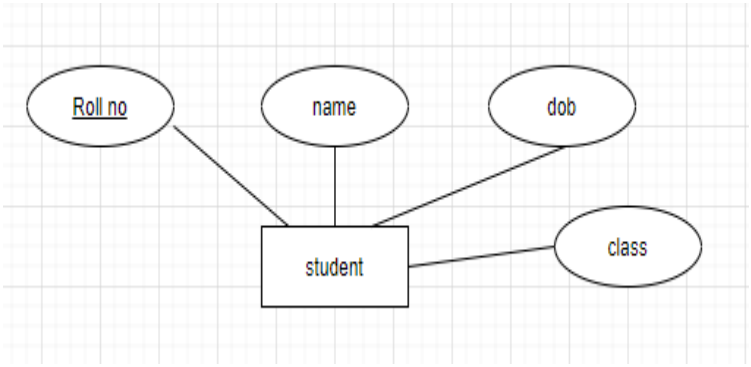


Database Management System
BIM SEM 4

I. Conversion of strong entity set with only simple attribute

a) Attribute of entity set will become attribute of table.
b) Key attribute of entity set will become primary key of table.

Example:



Student

Roll_no	Name	DOB	class

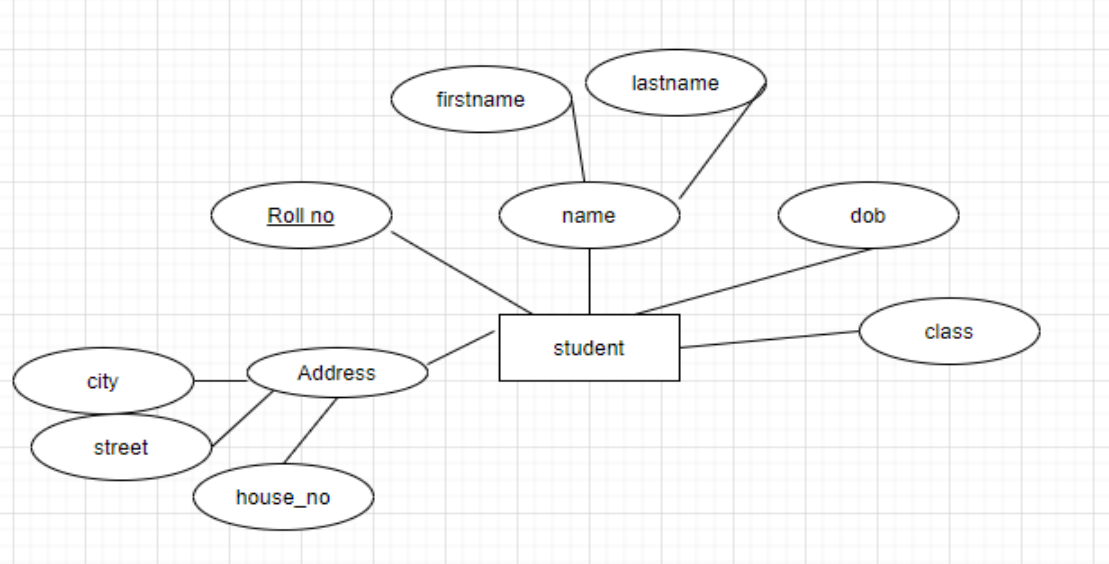
II. Conversion of strong entity set with composite attribute to relation

Composite Attribute

The attribute which can be decomposed into smaller sub attributes is called composited attributes. Composite attribute consists of two or more attributes.

Example:

In student entity the attribute name can be decomposed into first name and last name. Similarly, the address attribute can be decomposed into city, street and house no as shown below…



The above ER diagram can be converted into relation as follows…

Student

Roll_no	First_name	Last_name	House_no	street	city	dob	class

III. Conversion of strong entity set with multivalued attribute

Multivalued Attribute:

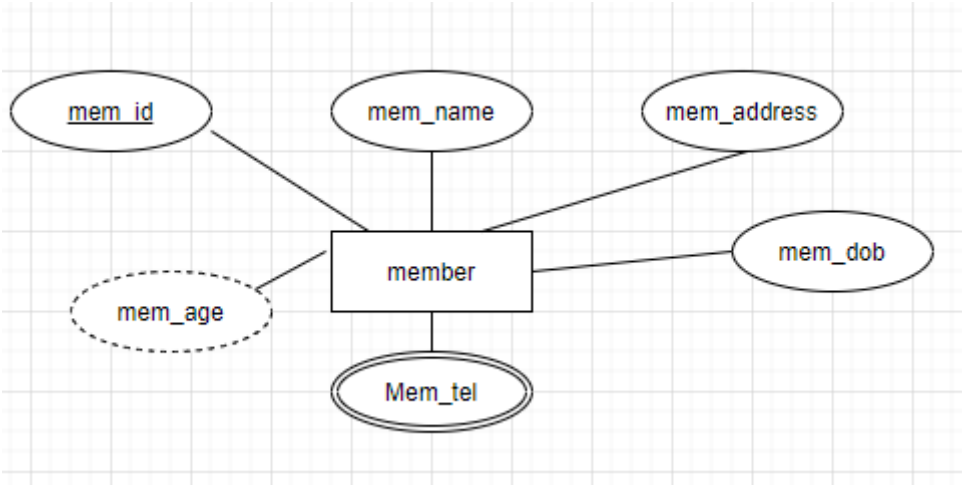
Multivalued attribute are attributes that can have many values. Example: telephone number of employee…

It can have separate home telephone, office telephone, mobile number.

When Strong entity set with any number of multivalued attributes is converted into relation there will be two table in relational model….

a) One table will contain all the simple attributes with the primary key
b) Other table will contain primary key and all multivalued attributes.

Example:



Above ER diagram can be converted into following tables

Member

Mem_id	Mem_name	Mem_adress	Mem_dob	Mem_age

Member-contact

Mem_id	**Tel_office**	**Tel_home**	**Mobile_no**

IV. Strong Entity set with Derived Attribute

Derived Attribute:

The attribute whose value may be calculated (derived) from other attributes is called derived attribute eg. Age (age can be calculated from date attribute)

So **derived attribute need not be physically store within the database** and can be derived by using an algorithm. After removing derived attribute age from above relation schema becomes….

Member

Mem_id	Mem_name	Mem_adress	Mem_dob

Database Management System
BIM SEM 4

❖ Representation of weak entity set

Weak entity set:

The entity set which does not have primary key is called weak entity set. It cannot be existed independently or it is dependent on a strong entity type. It is represented by **double rectangle.** The primary key of weak entity set is combination of partial key and primary key of strong entity set.

We can construct table of weak entity set is that the primary key from strong entity set and overall attribute from weak entity set.

Example:

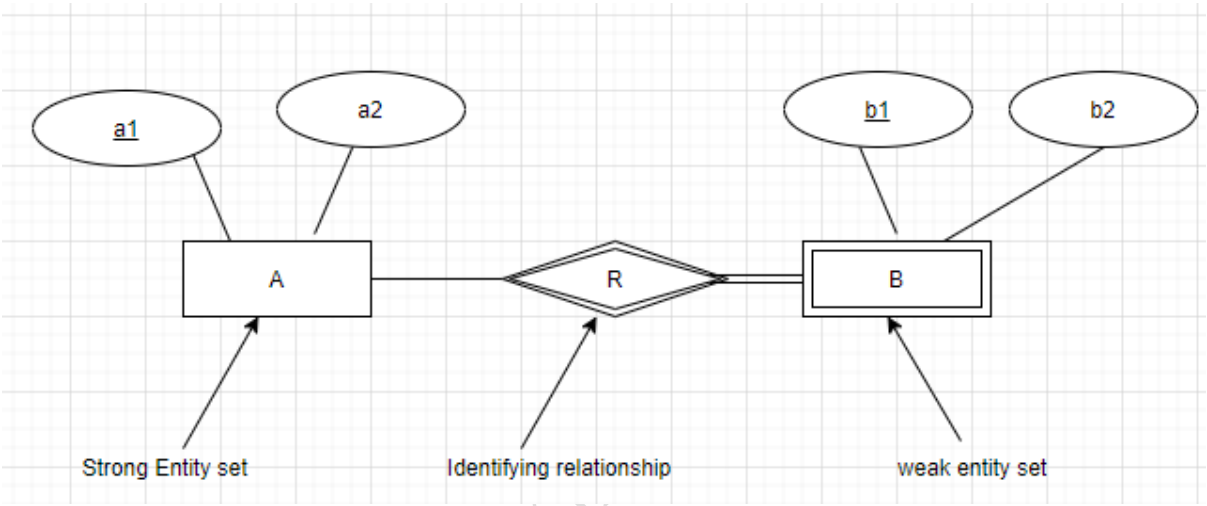


Table : A

a1	a2

Table : B

a1	b1	b2

Another Example:

Database Management System
BIM SEM 4

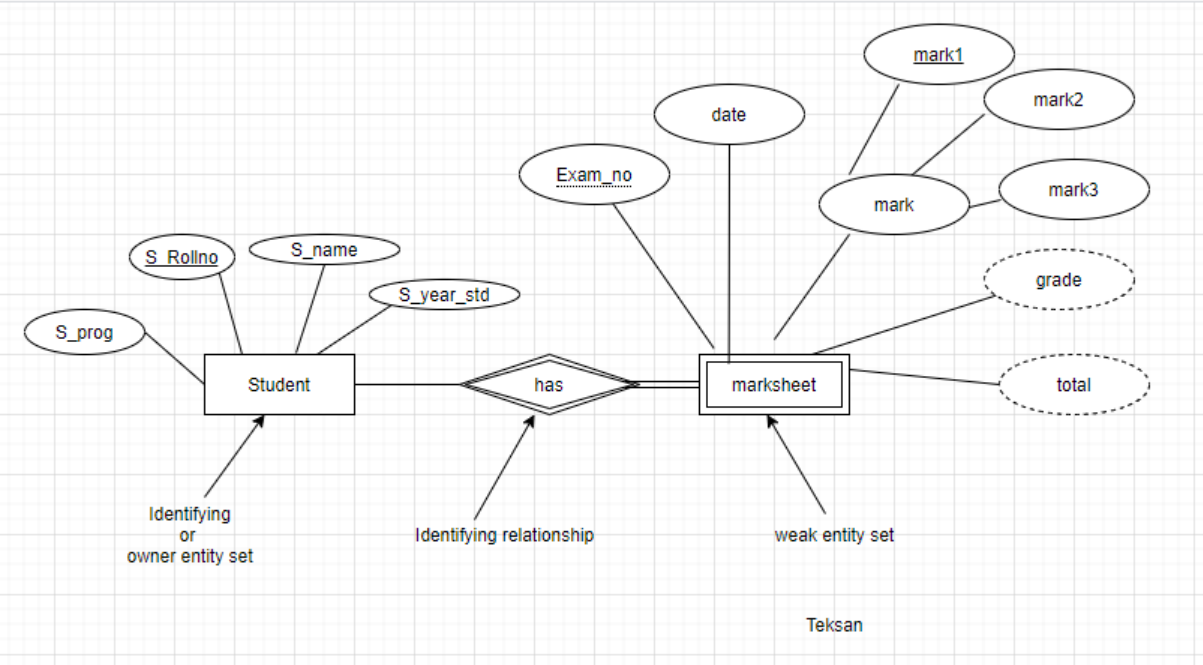


Table: Student

S_rollno	S_name	S_prog	S_year_std

Table: Marksheet

S_rollno (pk)	Exam_no (pk)	date	Mark1	Mark2	Mark3

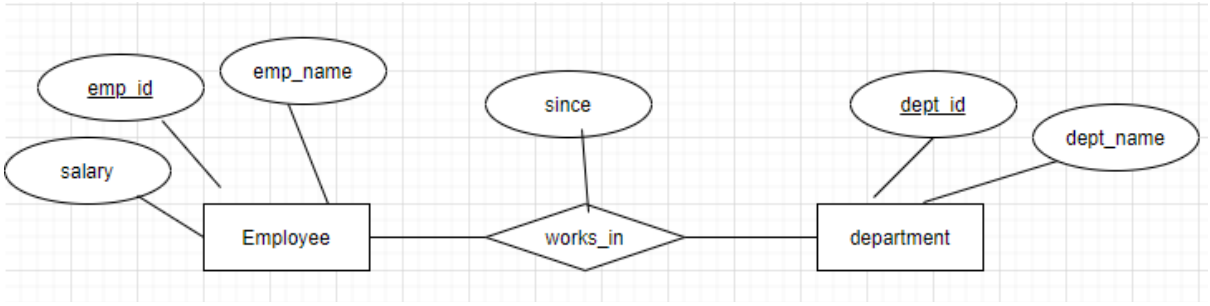
Note: Attribute total and garade are removed because they are derived attributes

Database Management System
BIM SEM 4

❖ Representation of relationship

A relationship set will require one table in the relational model. Attribute of table Will be

i) Primary key attribute of the participating entity set
ii) Its own descriptive attribute if any



Works_in

Emp_id	Dept_id	since

If we consider the overall ER diagram three table will be made in relational model they are

1) Employee table
2) Department table
3) Works_in table

During the translation of Relationship set into the relational model, it depends on the cardinality ratio. Following are the cases…

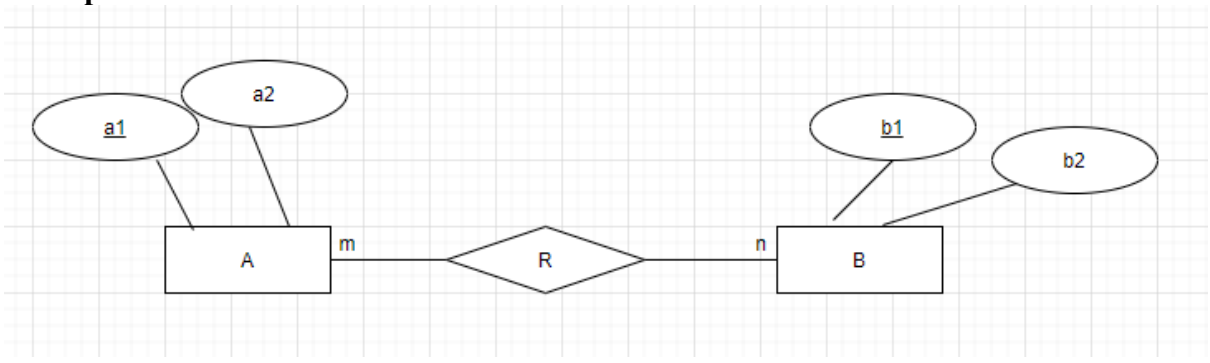
1. Binary relationship with cardinality ratio M:N
2. Binary relationship with cardinality ratio 1:N
3. Binary relationship with cardinality ratio M:1
4. Binary relationship with cardinality ratio 1:1

(Note: Here we will cover the syllabus, **Mapping Cardinalities: Representation of Mapping Cardinalities in E-R Diagram.)**

For Binary relationship with cardinality ratio M:N

We create a new relation a new relation and include the primary keys of the associated relations or tables as a composite key.

Example:



Database Management System
BIM SEM 4

There will be three table
Table : A

a1	a2

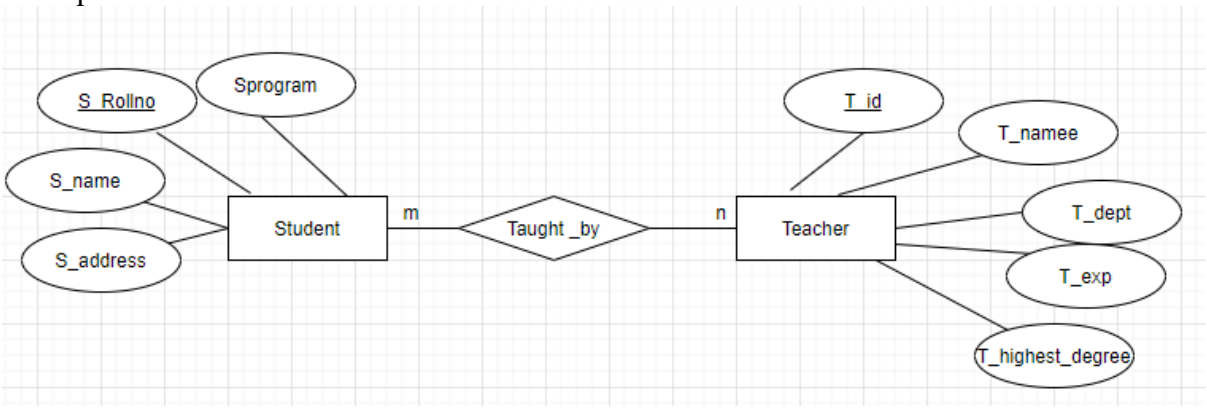
Table B:

b1	b2

Table : R

a1	b1

Example:



The tables are

Table : Taught _by

S_rollno	T_id

Table Student:

S_rollni	S_name	S_address	S_program

Table : Teacher

T_id	T_name	T_dept	T_exp	T_highisi_digree b1

smn

Database Management System
BIM SEM 4

For Binary relationship with cardinality ratio M:1

You shood add the primary key attribute (as attributes) of the entity on the **one side** of relationship as a foreign key in the relation on the **many sides.**

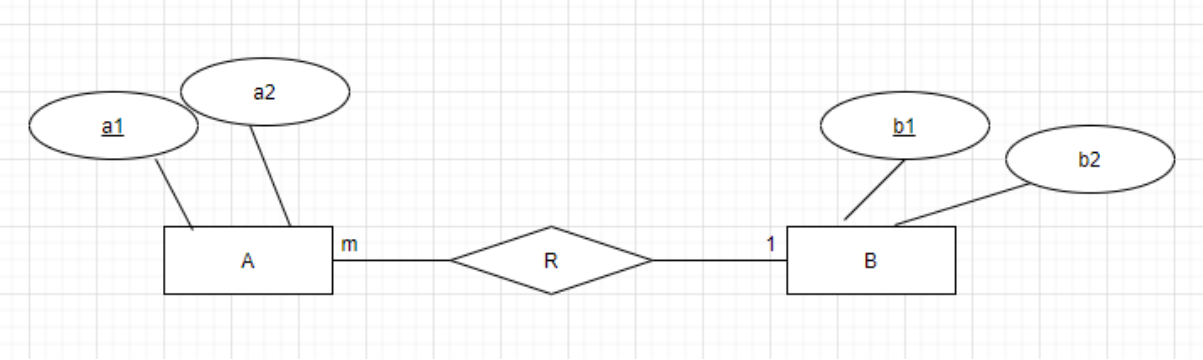


Table : AR

a1(pk)	a2	b1(fk)

Table B:

b1	b2

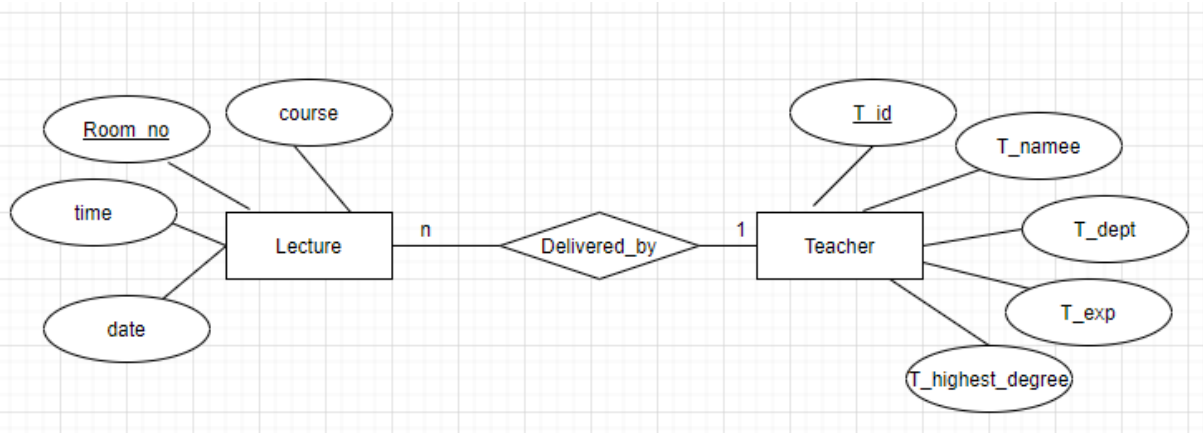


Table : Lecture

Room_no(pk)	time	course	date	T_id(fk)

smn

Database Management System
BIM SEM 4

Table :Teacher

T_id	T_name	T_dept	T_exp	T_higherdegree

For Binary relationship with cardinality ratio 1:M

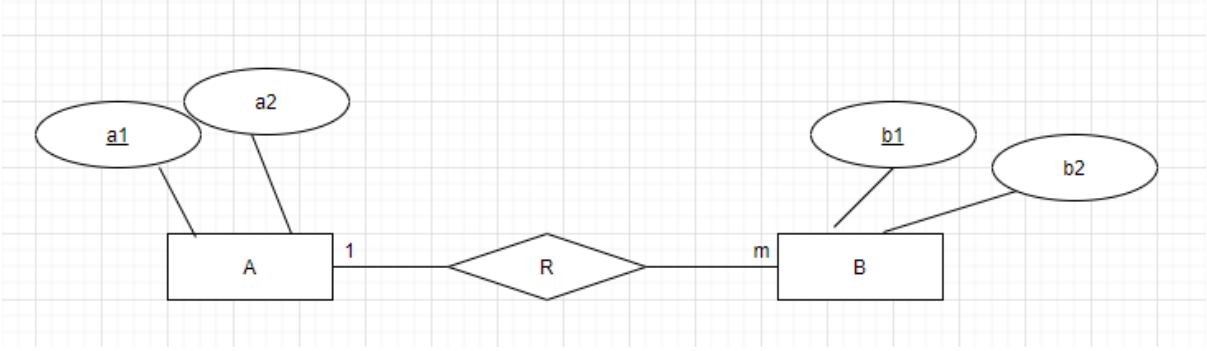


Table A:

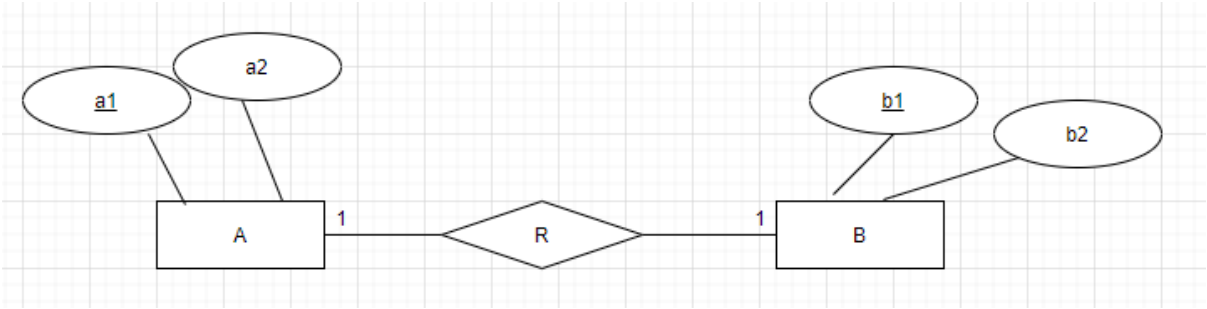
a1	a2

Table : BR

b1(pk)	b2	a1(fk)

For Binary relationship with cardinality ratio 1:1

There will require either Relation 'R' combine with table A and with table B



One way:

Table : AR

a1(pk)	a2	b1(fk)

smn

Database Management System
BIM SEM 4

Table B:

b1	b2

Another Way:

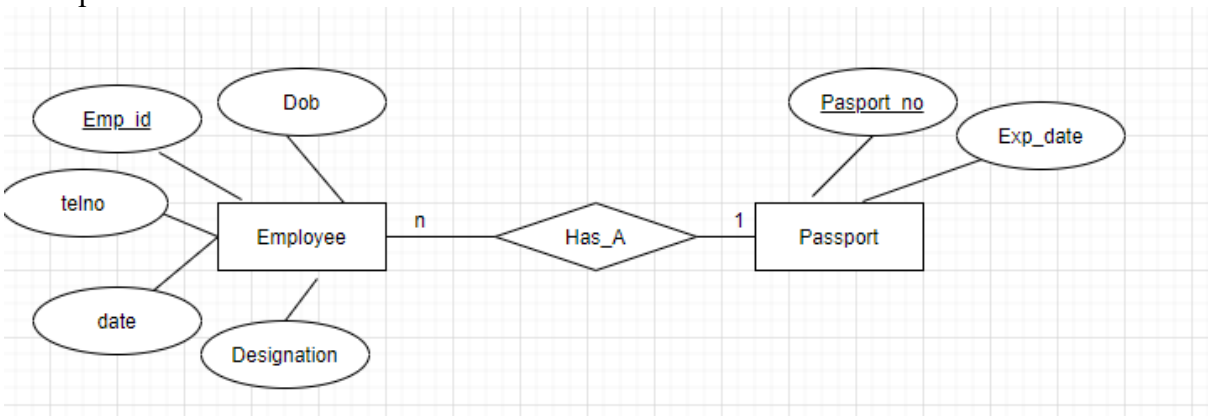
Table A:

a1	a2

Table : BR

b1(pk)	b2	a1(fk)

Example:



One way:

Table : Employee

Emp_id(pk)	name	designation	Tel_no	dob	Passport_no(fk)

Table :Passport

Passport_no	Exp_date

Another Way:

Table :Passport

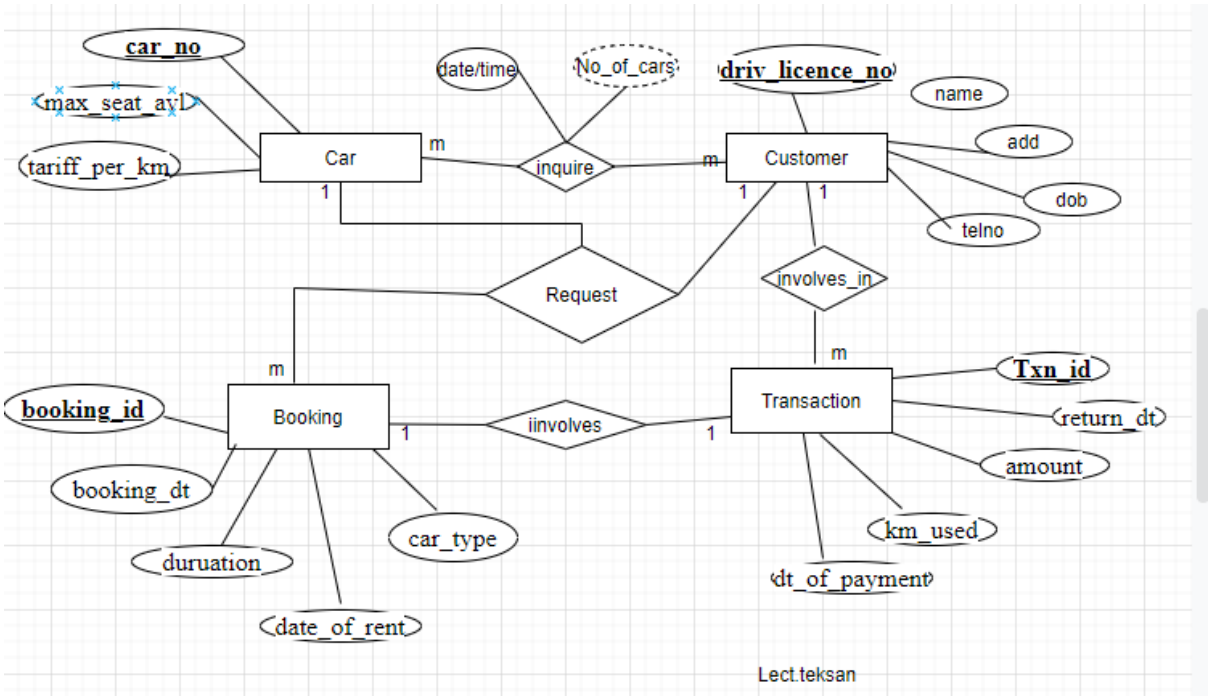
Passport_no	Exp_date	Emp_id

Database Management System
BIM SEM 4

Table : Employee

Emp_id(pk)	name	designation	Tel_no	dob

Example: Case studies (Subu-car rental service)



Solution:

Customer

Driv_licence_no	name	dob	address	Tel_no

Transaction

Txn_id	Return_dt	Km_used	Dt_of_payment	Amount	Driving_licence_no

Car

Car_no	Max_seat_avl	Tariff_per_km	Car_type

Inquire

Driving_licence_no	Date/time	No_of_cars	Car_no

smn

Database Management System
BIM SEM 4

Booking

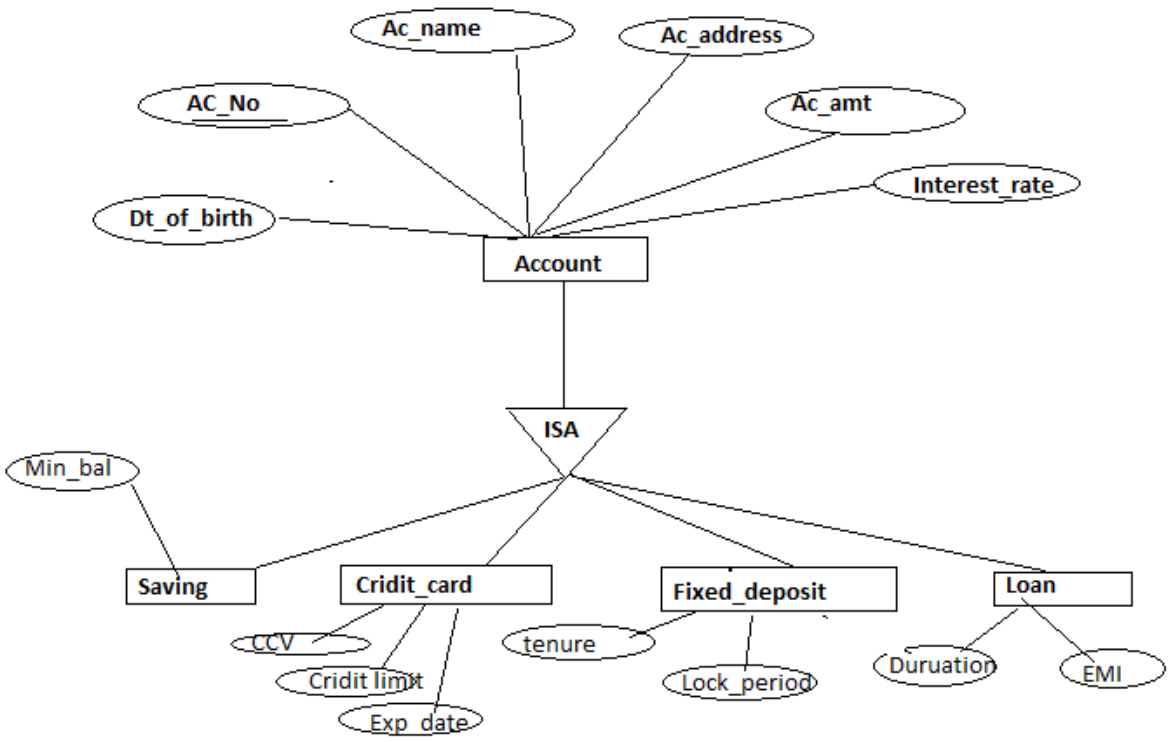
Booking_id	Driv_licence_no	Car_no	Txn_id	Book_dt	Dt_of_rtn	duruation

Specialization

- Specialization is a process by which an entity set is decomposed into subgroups consisting of lower entities which inherits the attributes of the higher-level entities.
- It is a top-down approach, in which one higher entity can be broken down into two lower-level entity.
- It defines one or more sub class for the super class and also forms the superclass/subclass relationship.

Example:-

To illustrate the concept we can consider an Account which can be of various type like “saving”, “credit card”, ”fixed deposit” and loan. The processes is depicted by a triangle composed of ISA labelled.



The attribute of parent entity set “Account” is inherited by the child sub entities and is common to all the entities “saving”, “credit card”,” fixed deposit” and loan. But besides these general attributes each sub entity has specialised attribute which distinguishes that entity from other entity.

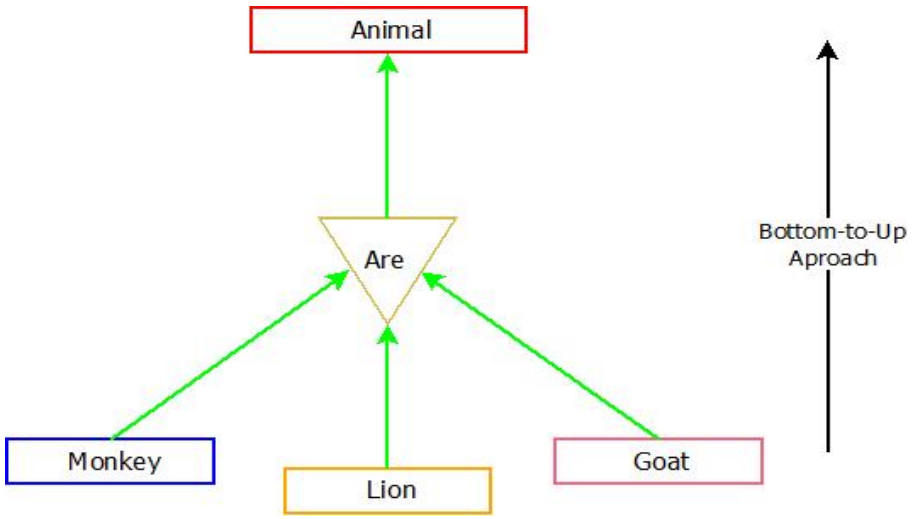
Database Management System
BIM SEM 4

Generalization

Generalization is the reverse process of specialization. In generalization, the sub entities are combined together into a super entity set on the basis of some common features in such a way that the new entity thus formed contains all the features of the sub entities. Generalization is a Bottom-to-Up approach.

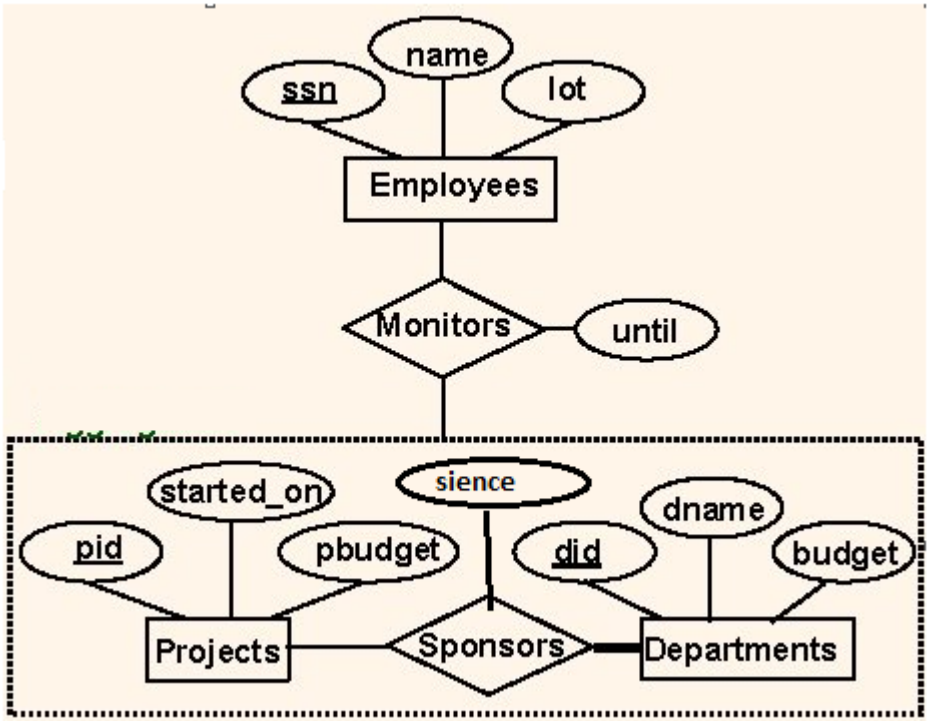
For example:

Goat, Lion and Monkey have some come traits and they all together can be combined in class of animals.



Aggregation:

Aggregation allows us to indicate that a relationship set (identified through a dashed box) participates in another relationship set.



- Suppose that we have an entity set called Projects and that each Projects entity is sponsored by one or more departments. The Sponsors relationship set captures this all information.

- A department that sponsors a project might assign employees to monitor the sponsorship. Intuitively, Monitors should be a relationship set that associates a Sponsors relationship (rather than a Projects or Departments entity) with an employee's entity. Such type of relationship is called aggregation. Which is illustrated in above figure with dashed box around sponsors used to denote aggregation.

- Aggregation can be used when we need to express a relationship among relationships. In our example, why we not make Sponsors a ternary relationship? The answer is that there are really two distinct relationships, Sponsors and Monitors, each possibly with attributes of its own.

- For instance, the Monitors relationship has an attribute until that records the date until when the employee is appointed as the sponsorship monitor. Compare this attribute with the attribute since of Sponsors, which is the date when the sponsorship took effect.

EER Model

The Er model views the real worlds as a collection of objects or entities and the relationship among them. The ER modelling concept can model only traditional system.

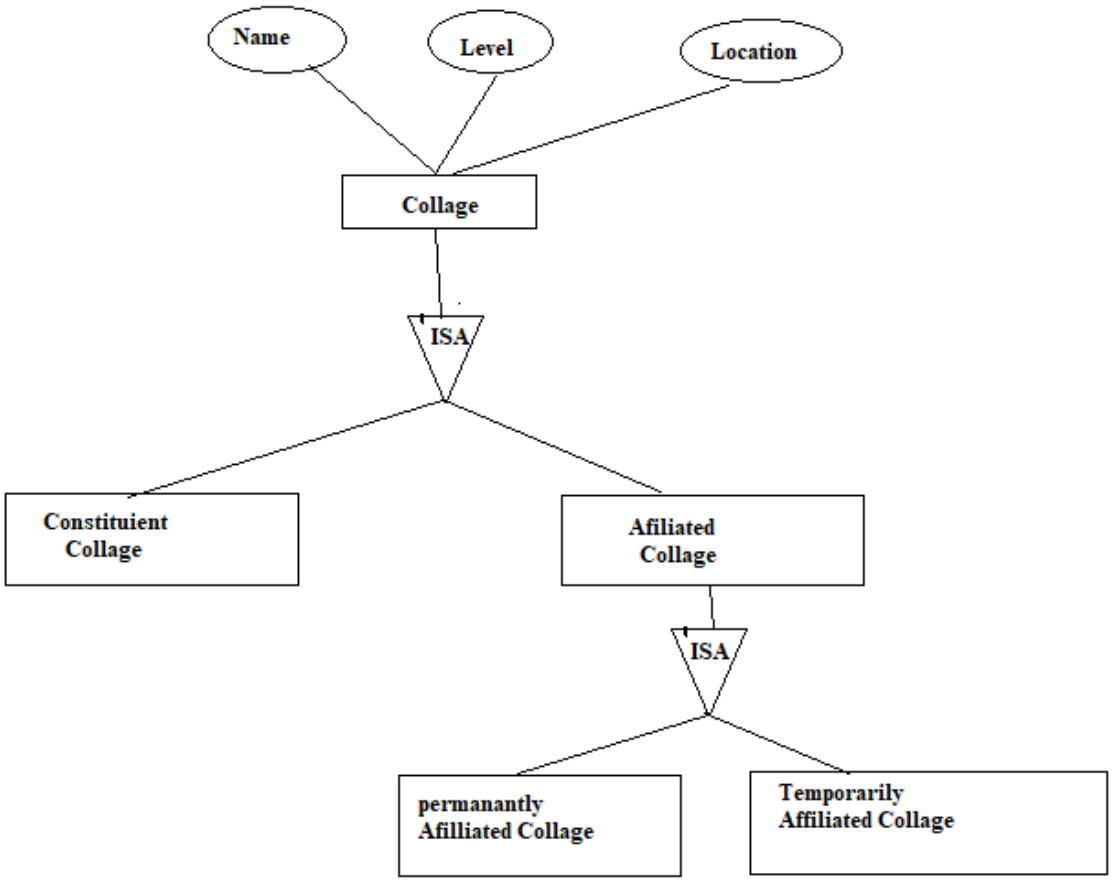
So, in order to model modern complex system such as database for engineering design and manufacturing, telecommunication as complex software system and geographical information system (GIS) there is needed of extended ER model.

EER is a high-level data model that incorporates the extension to the ER model. It is also called **enhanced ER model**. It includes all of the concept introduced by the ER model. In addition to Er model EER model includes…

1. Subclass and super class
2. Specialization and generalization
3. Category or Union type
4. Aggregation.

Subclass and Super class

Super class is an entity that has a relationship with one or more subtype. Subclass is a group of entities with unique attribute and it inherits the properties and attribute from its super class. In ER diagram we can represent super class / sub class relationship by using ISA triangle as follows.



In above diagram, **Collage** entity set is super class, **Constituent collage** and **Affiliated collage** entity are sub class of super class entity set Collage. **Further Permanently affiliated collage** and **temporarily affiliated** collage entity set are sub class of **Affiliated collage.**

Category Or Union Type

Category represent a single super class or subclass relationship with more than one super class. It can have total or partial participation. For example

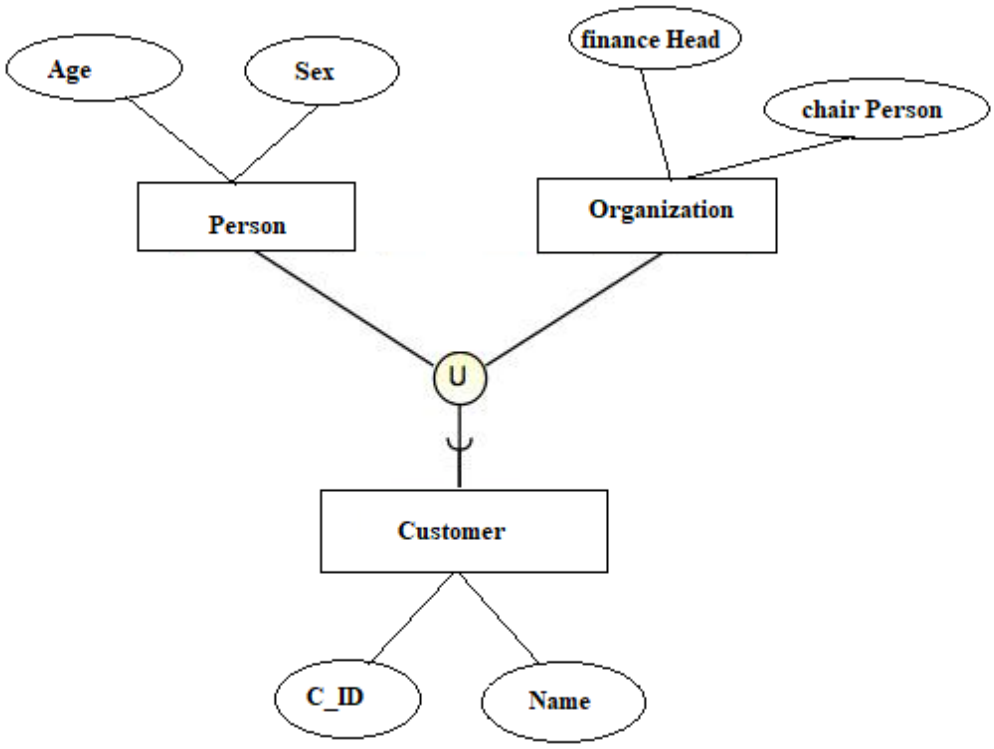


Figure: Category

In above figure customer in the bank may be organization or it may be person but not both. Hence Customer is a subset of the union of two super class person and organization entity set.

A category member must be existing in at least one of its super classes.

Another example:

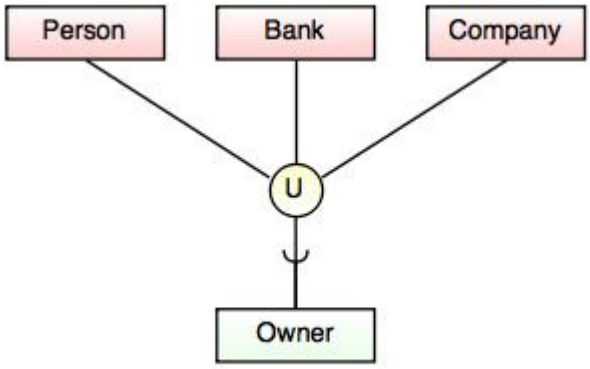


Fig. Categories (Union Type)

For example, Car booking, Car owner can be a person, can be bank or can be company. Owner (Category or sub class) is a subset of the union of the three super classes Company, Bank, and Person. A Category member must exist in at least one of its super classes.

========== End unit 2 ==========

smn

Database Management System
BIM SEM 4

Unit 2: Entity-Relationship Model (LH 8)

Entities and Entity Sets. Relationships and Relationship Sets. Attributes. Mapping Constraints. Keys (Super key, Candidate key and Primary key): Primary Keys for Entity Sets and Relationship Sets. The Entity Relationship Diagram. Reducing E-R Diagrams to Tables: Representation of Strong Entity Sets, Representation of Weak Entity Sets, Representation of Relationship Sets. Generalization and Specialization. Aggregation. Mapping Cardinalities: Representation of Mapping Cardinalities in E-R Diagram. Use of Entity or Relationship Sets. Use of Extended E-R Features. Design of an E-R Database Scheme (Case study).

- ⇨ The Entity Relationship Diagram. Reducing E-R Diagrams to Tables: Representation of Strong Entity Sets, Representation of Weak Entity Sets, Representation of Relationship Sets. Generalization and Specialization. Aggregation.
- ⇨ Mapping Cardinalities: Representation of Mapping Cardinalities in E-R Diagram.
- ⇨ Use of Entity or Relationship Sets.
- ⇨ Use of Extended E-R Features.
- ⇨ Design of an E-R Database Scheme (Case study).