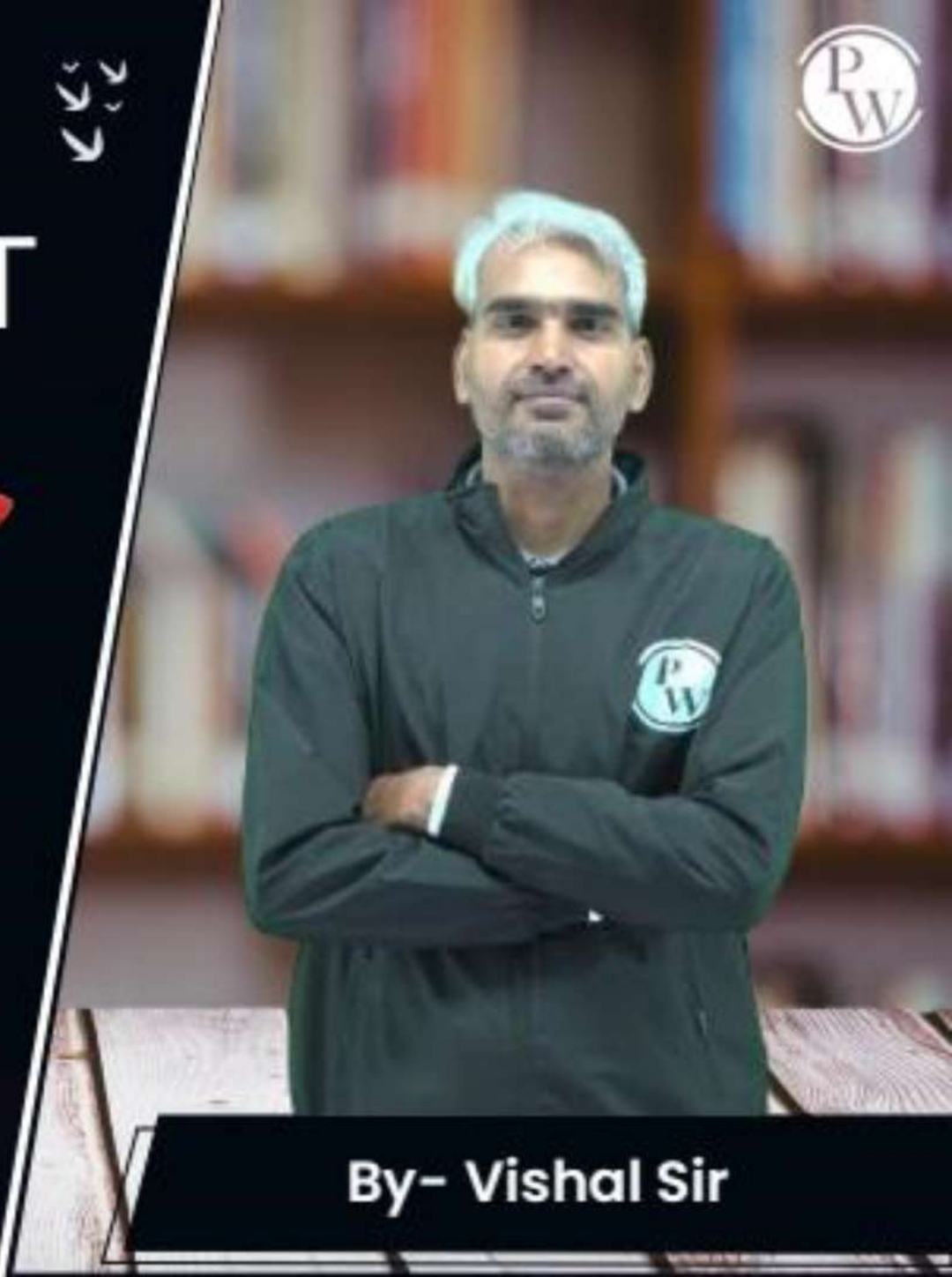
Computer Science & IT

Database Management System

Entity Relationship Model & Integrity constraints

Lecture No. 02



By- Vishal Sir

Recap of Previous Lecture

Topic ER model & ER diagram

Topic Relational model & Integrity constraints

Slide

Topics to be Covered

- **Topic** ER model & ER diagram
- **Topic** Relational model & Integrity constraints

Slide

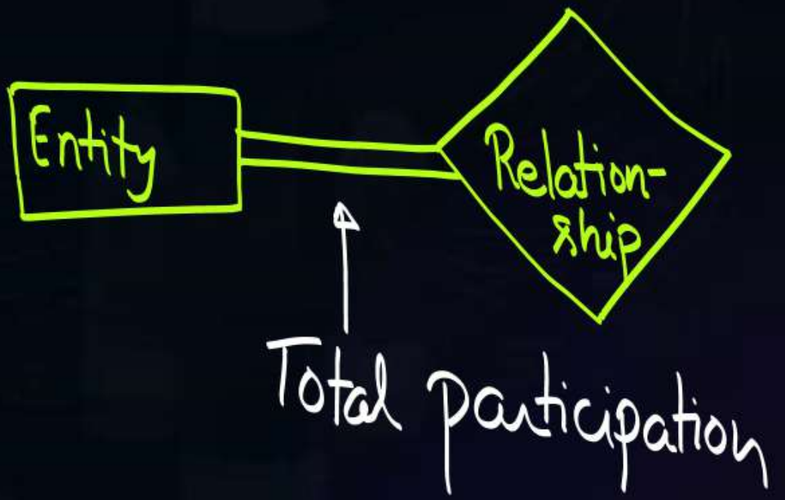
Topic : Participation constraints

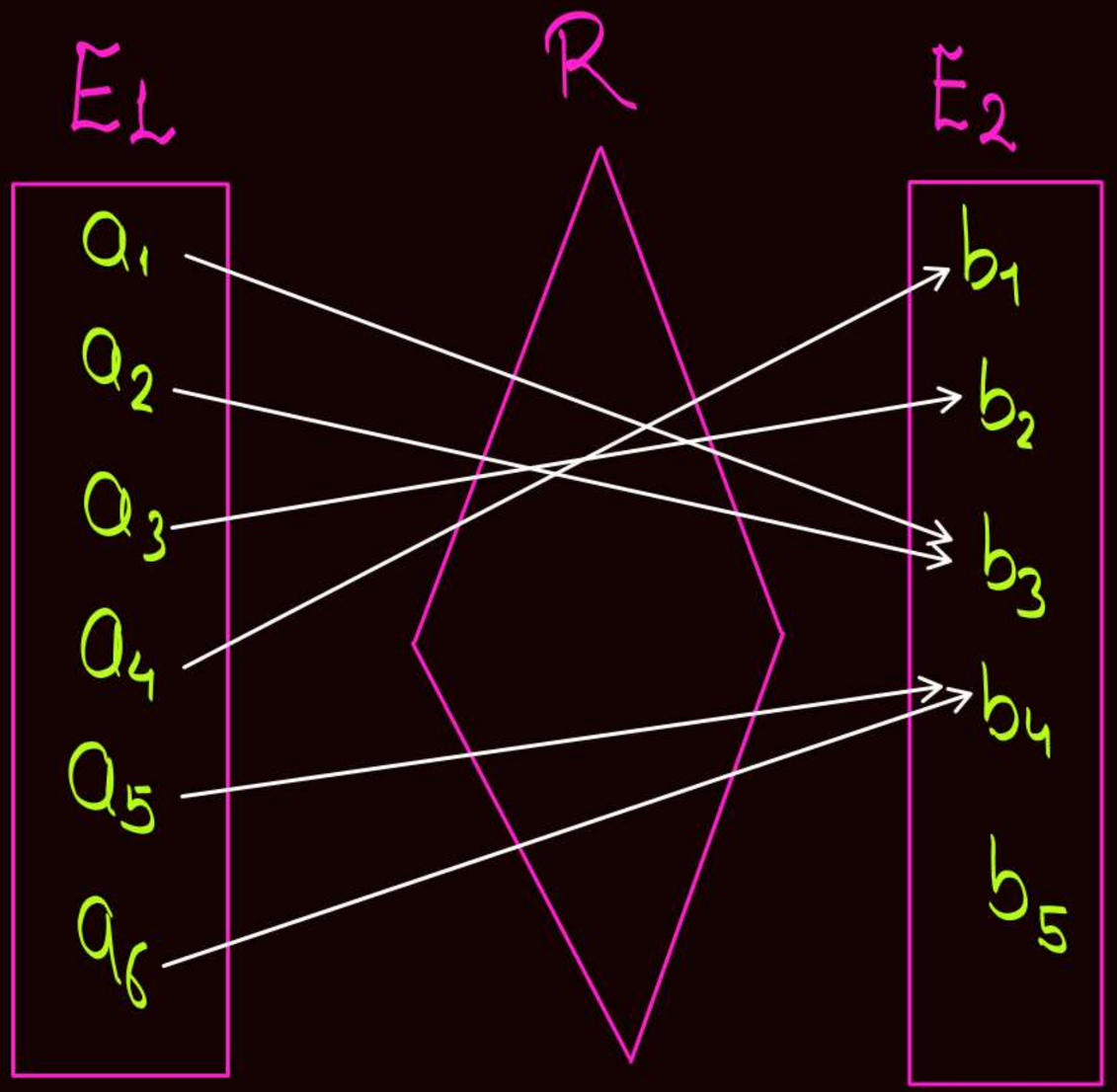
There are two types of participation constraints :

1. Total participation
2. Partial participation

Topic : Total participation

- If every entity of an entity set participate in the relationship set, then that entity set is said to have total participation.
- Total participation of an entity set in a relationship set is denoted by double line



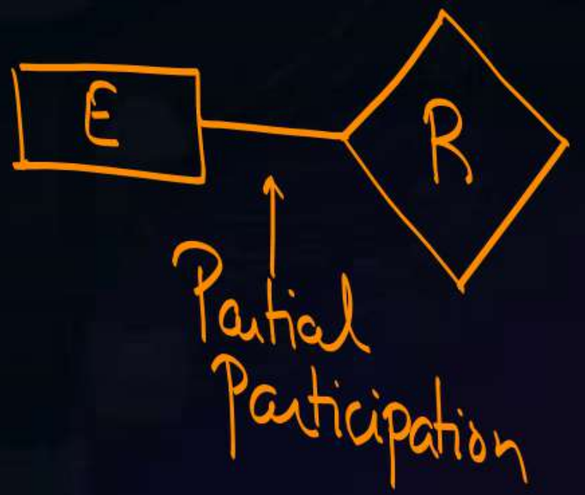


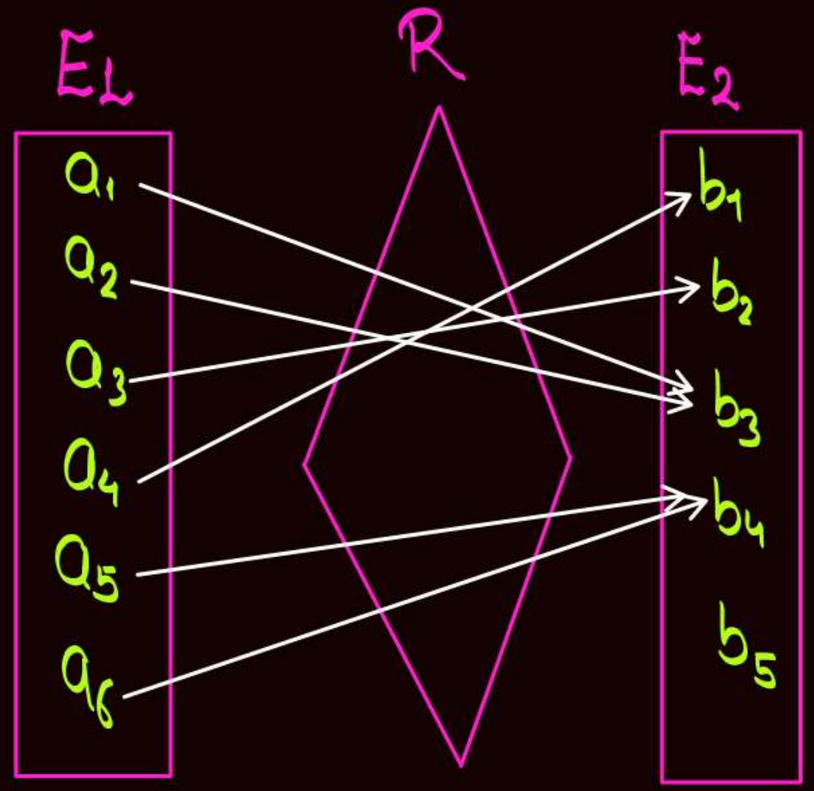
Each entity of entity set E_1 participate in relationship set R.

∴ Entity set E_1 has total participation w.r.t. relationship set R

Topic : Partial participation

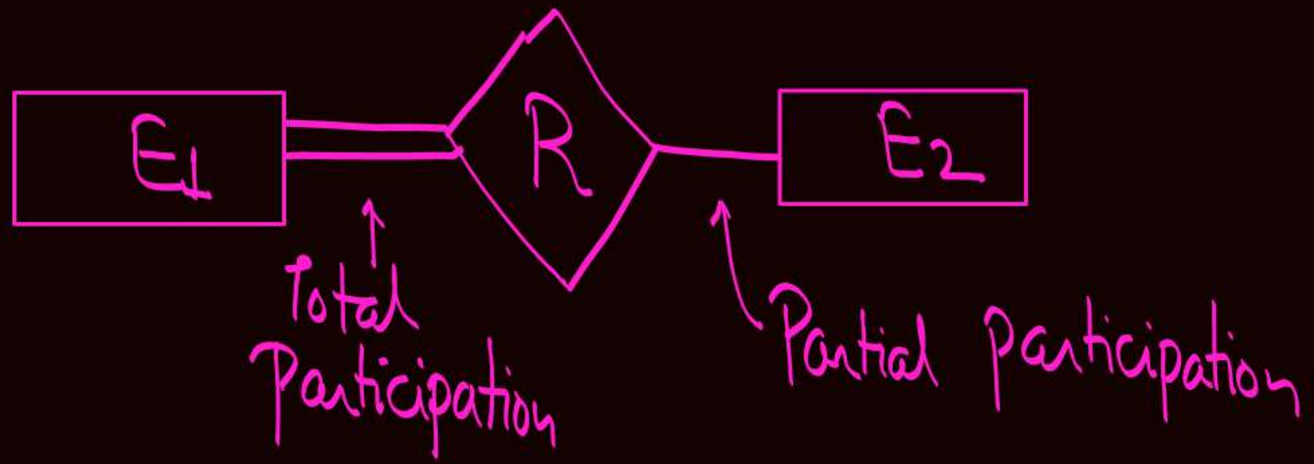
- If at least one entity of entity set is not participating in the relationship set, then that entity set is said to have partial participation.
- Partial participation of an entity set in a relationship set is denoted by single line.

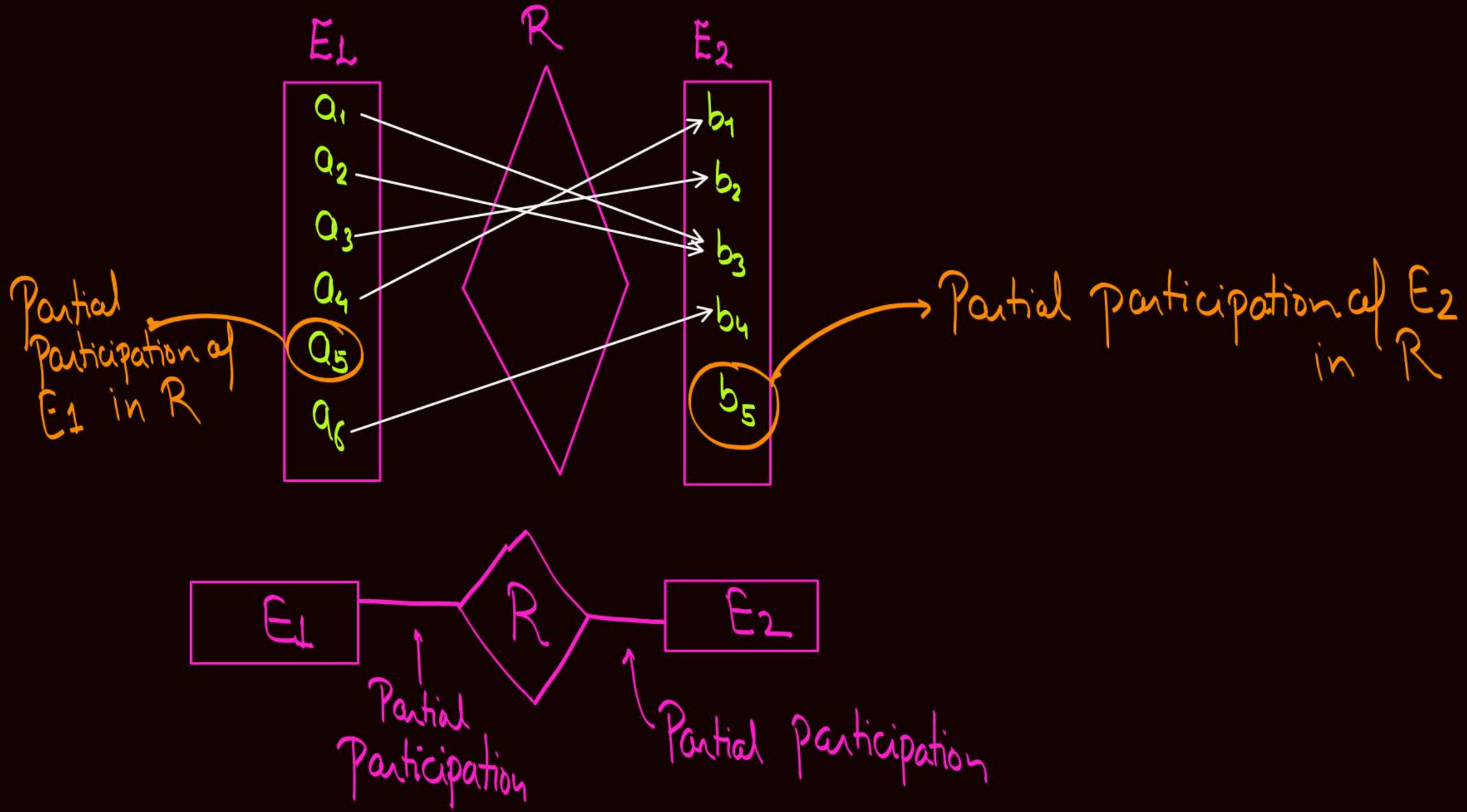


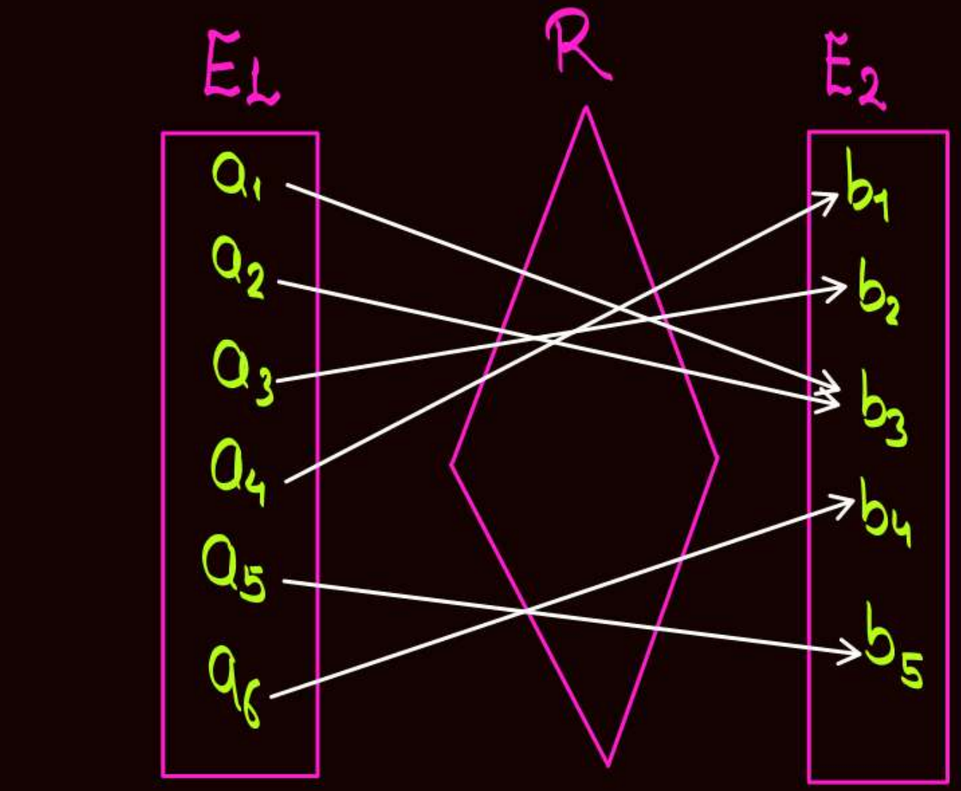


* Entity "b_5" of entity set E_2 does not participate in relationship set R.

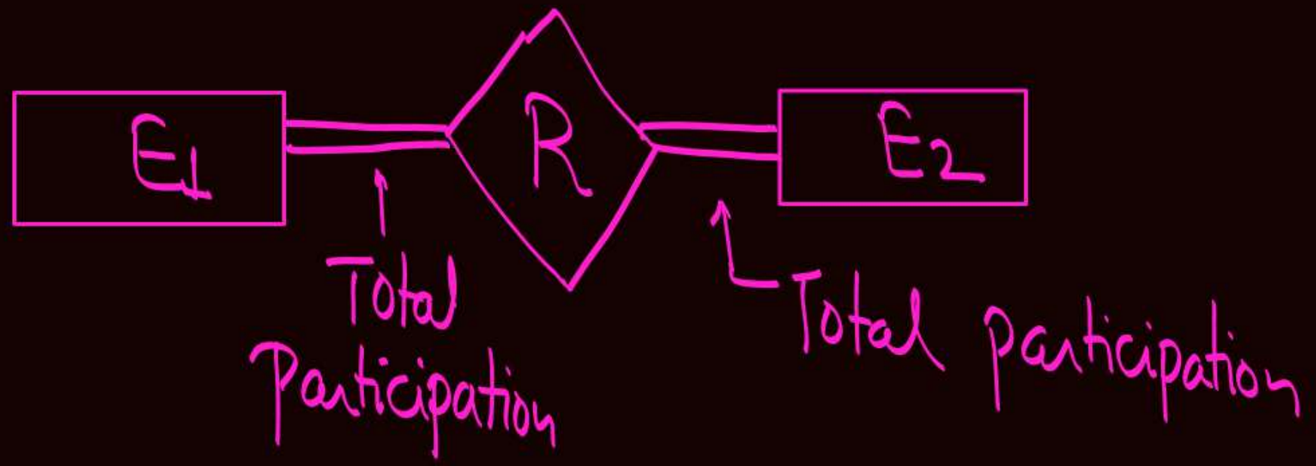
∴ Entity set E_2 has Partial participation w.r.t. relationship set R.




E1
R
E2
a1
a2
a3
a4
a5
a6
b1
b2
b3
b4
b5
Partial Participation of E1 in R
Partial participation of E2 in R
E1
R
E2
Partial Participation
Partial participation



Both E1 and E2 has total participation with relationship set R



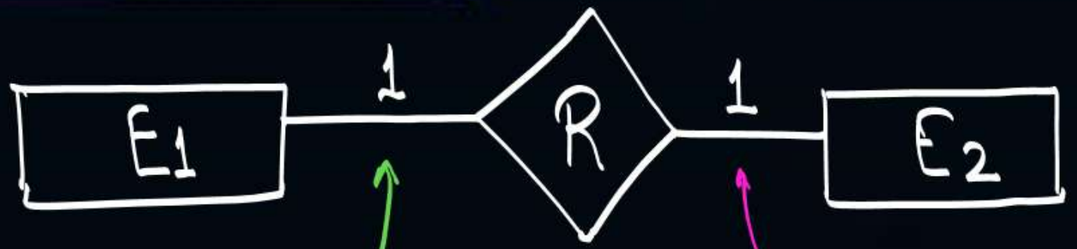
Topic : Mapping cardinality

Mapping cardinality or cardinality ratio is used to denote the number of entities to which another entity can be associated through a certain relation set.

For a binary relationship set mapping cardinalities must be one of the following types:

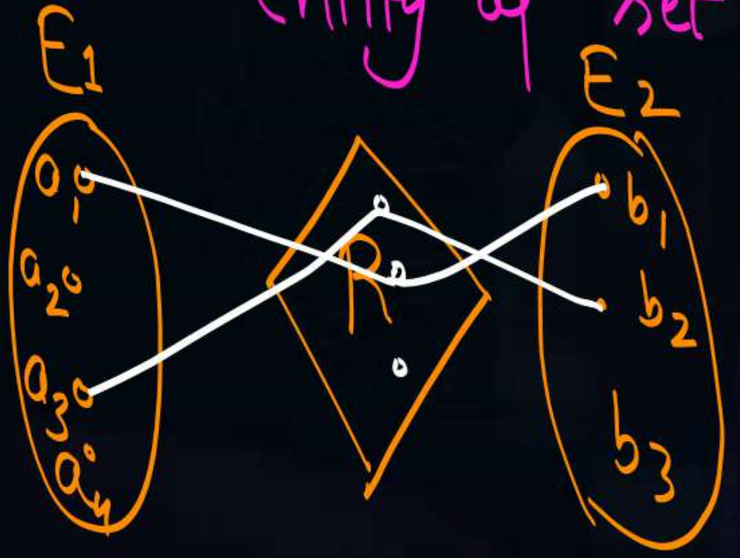
1. One-to-one (1 : 1)
2. One-to-many (1 : N)
3. Many-to-one (N : 1)
4. Many-to many (N : M)

Topic : One-to-one



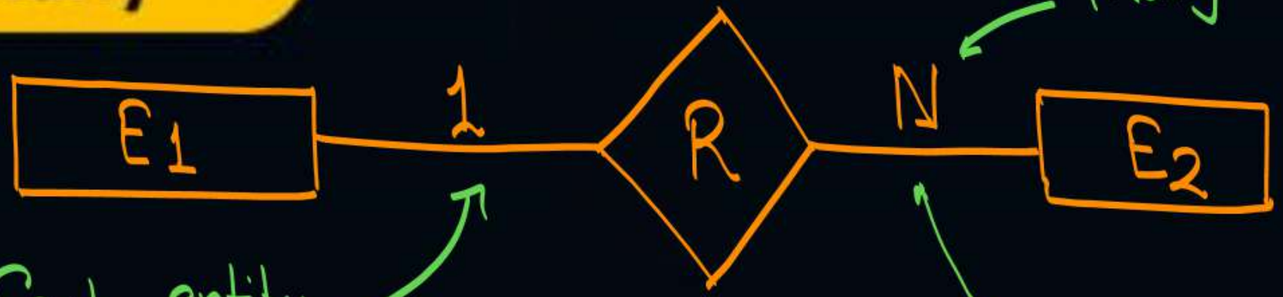
Entity of set E_2 Can associate with at most one entity of set E_1

Entity of set E_1 Can association with atmost one entity of set E_2



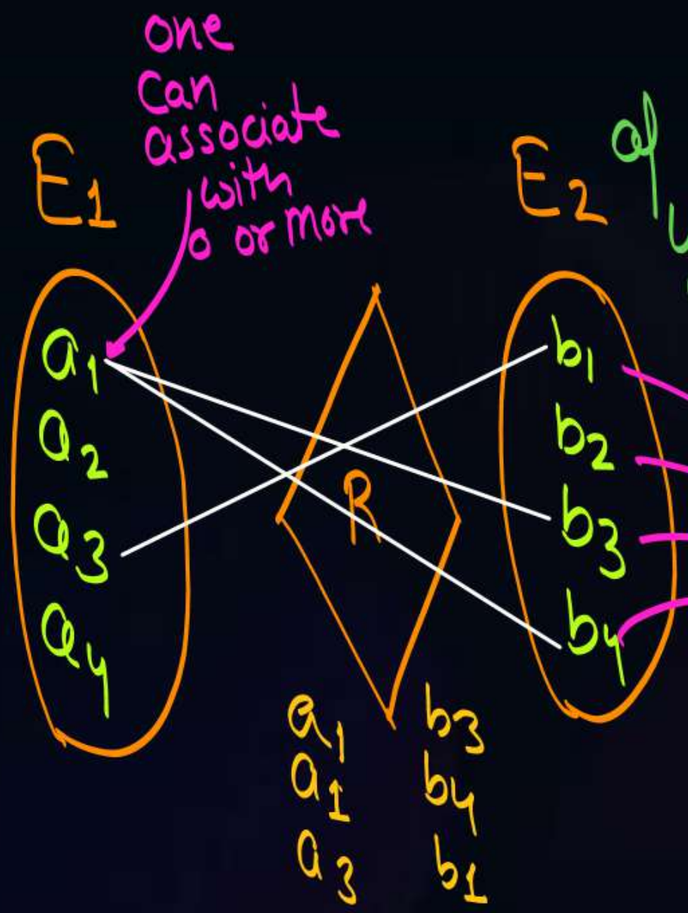
Topic : One-to-many

$E_1 : E_2 = 1 : N$



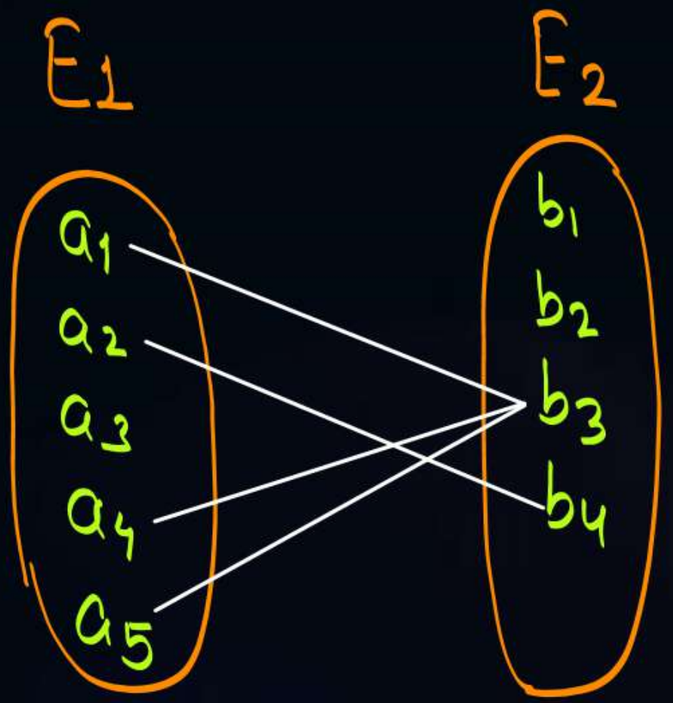
Each entity of set E_2 can associate with atmost one entity of set E_1

Each entity of set E_1 can associate with at most N {i.e. 0 to N} {0 or More} entities of set E_2



one can associate with at most '1'

Topic : Many-to-one

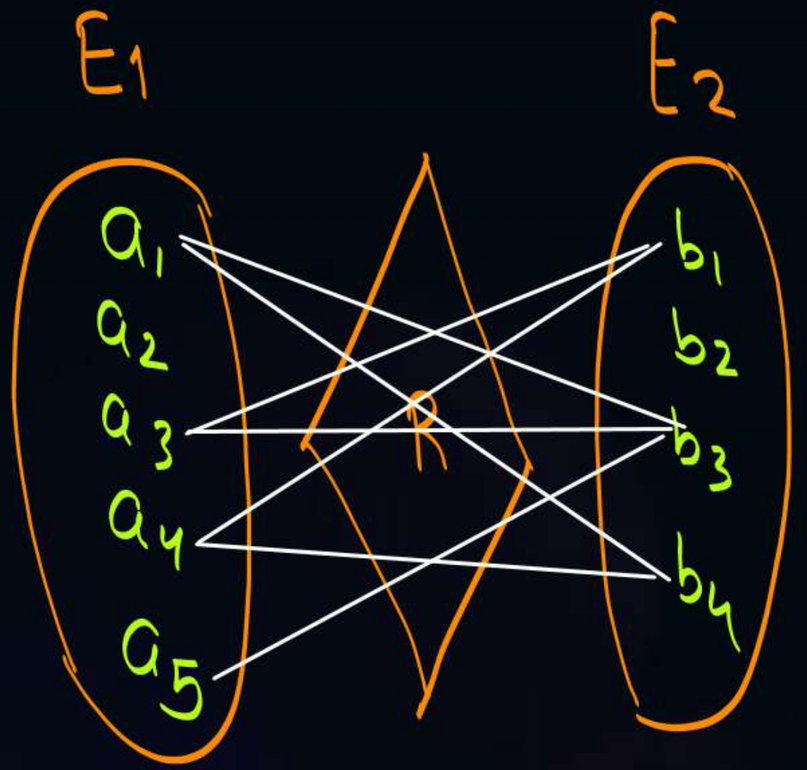


$E_1 : E_2 \equiv N : 1$

Each entity of set E_2 Can associate with 0 or More (N) entities of set E_1

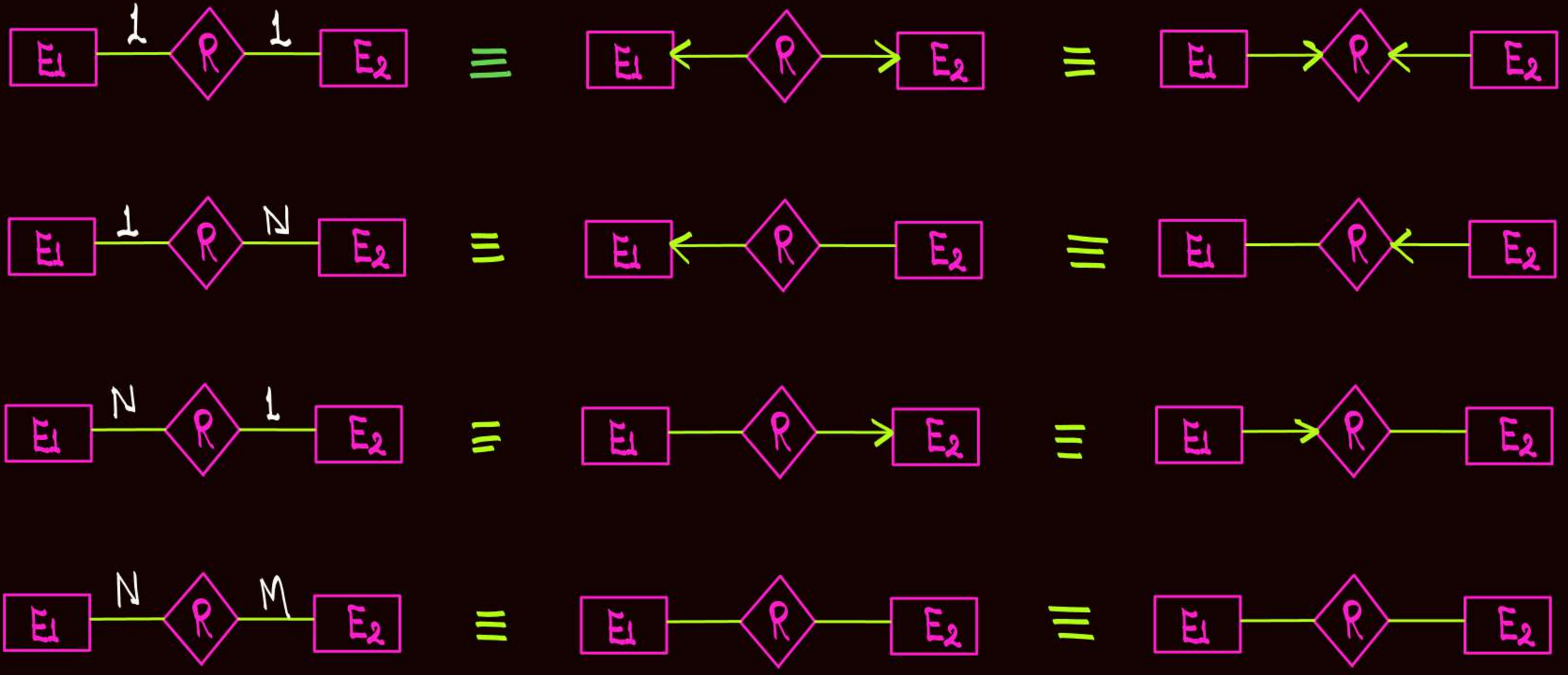
Each entity of set E_1 can associate with at most one entity of set E_2

Topic : Many-to-many



Each entity of set E_2 can associate with 0 or more entities of set E_1

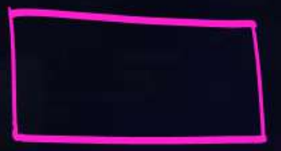
Each entity of set E_1 can associate with 0 or more entities of set E_2


E1
1
R
1
E2
≡
E1
R
E2
≡
E1
R
E2
E1
1
R
N
E2
≡
E1
R
E2
≡
E1
R
E2
E1
N
R
1
E2
≡
E1
R
E2
≡
E1
R
E2
E1
N
R
M
E2
≡
E1
R
E2
≡
E1
R
E2

Topic : Strong entity Set

- A strong entity set is an entity set that contains sufficient attributes to uniquely identify all its entities. In other words, a primary key exists for a strong entity set.
- Primary key of a strong entity set is represented by underlining it.

Strong Entity set are represented using single line rectangle



Topic : Weak entity Set

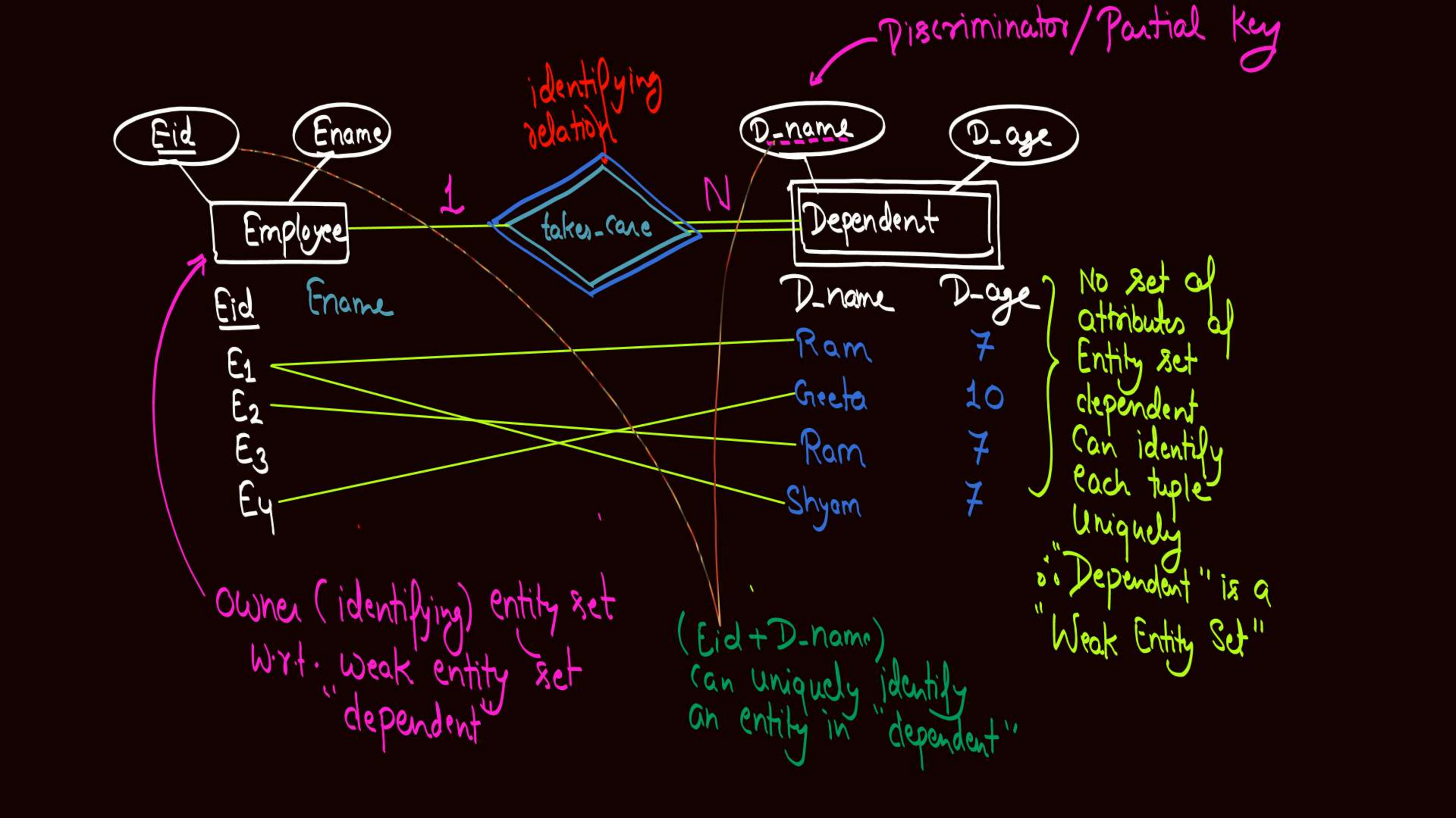
- A weak entity set is an entity set that does not contain sufficient attributes to uniquely identify its entities.
- In other words, a primary key does not exist for a weak entity set.
- However, it contains a partial key called as a discriminator.
- Discriminator can identify a group of entities from the entity set.
- Discriminator is represented by underlining with a dashed line.
- A weak entity must participate in an identifying relationship type with an owner or identifying entity type

Topic : Note

- A weak entity can be identified uniquely only by combining partial key of weak entity set with primary key of owner entity.
- Owner entity set and weak entity set must participate in an one-to-many relationship set. (Owner side is 1 and weak entity side is many)
- Weak entity set must have total participation in identifying relationship set.

① Weak entity set is represented using double line rectangle

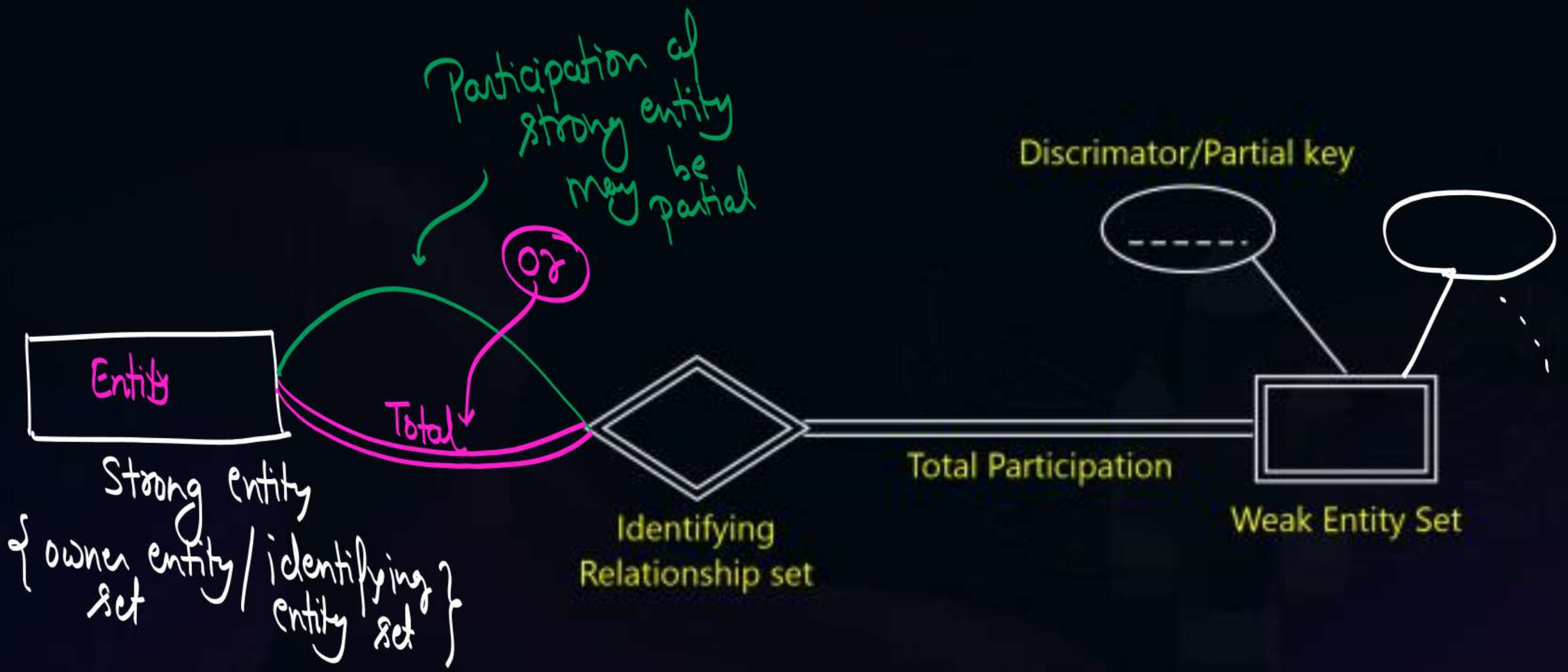
② Identifying relation will be represented using double line diamond


Discriminator / Partial Key
identifying relation
Eid
Ename
D_name
D_age
1
N
Employee
takes_care
Dependent
Eid
Ename
E1
E2
E3
E4
D_name
D_age
Ram
7
Geeta
10
Ram
7
Shyam
7
No set of attributes of Entity set dependent can identify each tuple Uniquely
∴ "Dependent" is a "Weak Entity Set"
Owner (identifying) entity set w.r.t. weak entity set "dependent"
(Eid + D_name) can uniquely identify an entity in "dependent"

Topic : Differences between Strong entity set and Weak entity set

Strong entity set	Weak entity set
A single line rectangle is used for the representation of a strong entity set.	A double line rectangle is used for the representation of a weak entity set.
It contains sufficient attributes to form its primary key.	It does not contain sufficient attributes to form its primary key.
A single line diamond symbol is used for the representation of the relationship that exists between the two strong entity sets.	A double line diamond symbol is used for the representation of the identifying relationship that exists between the strong and weak entity set.
Total participation may or may not exist in the relationship.	Total participation always exists in the identifying relationship.

Topic : Weak entity set in ER diagram



Min-Max Representation

Participation is represented using order pair of type (min, max)

1st value will be minimum value

2nd value will be maximum value

- Minimum number of times an entity can appear in the relationship set is represented by "min"

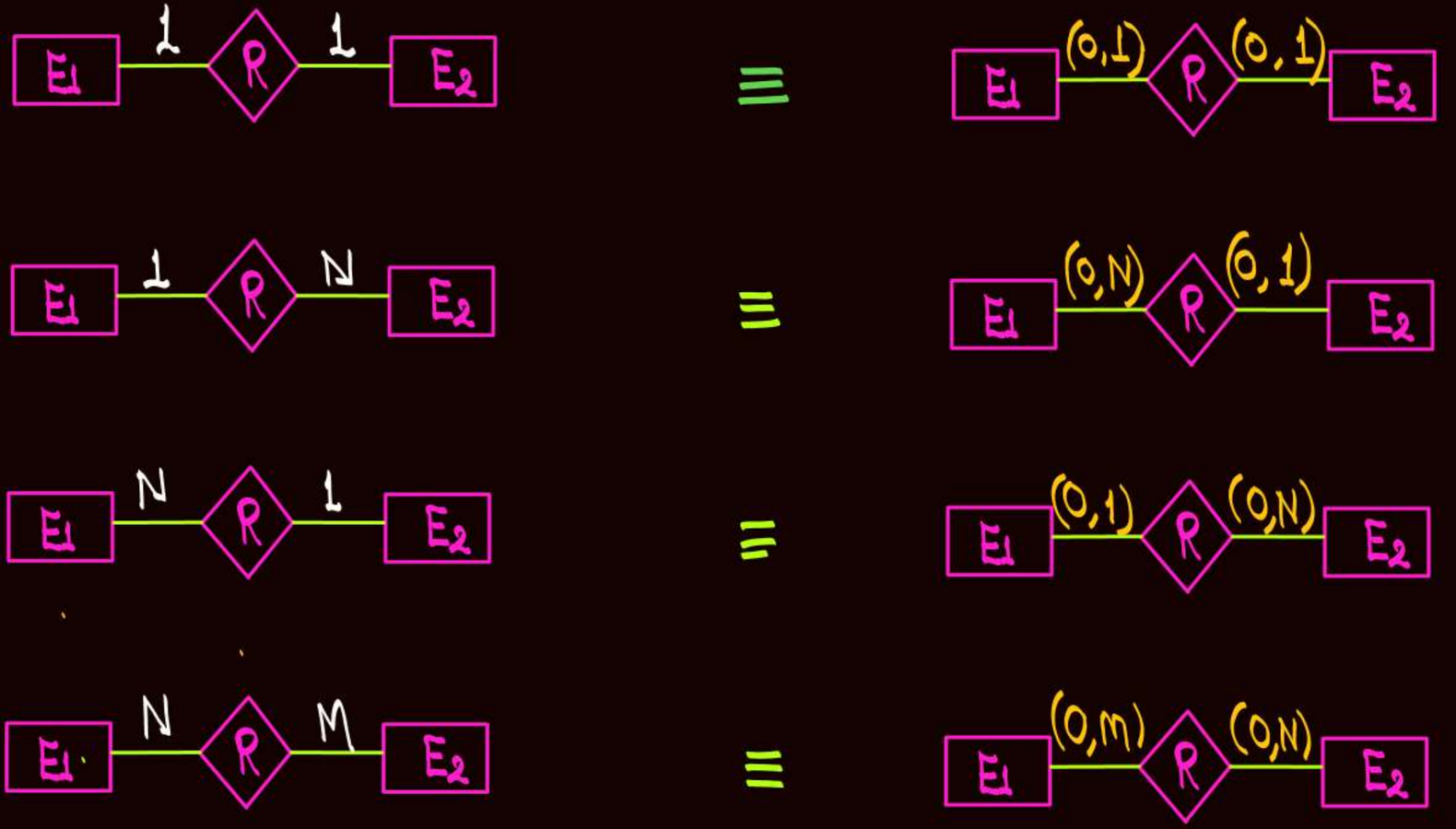
and " Maximum number of times an entity can appear in the relationship set is represented by "max"

Min-Max Representation

Participation is represented using order pair of type (min, max)

1st value will be minimum value

2nd value will be maximum value

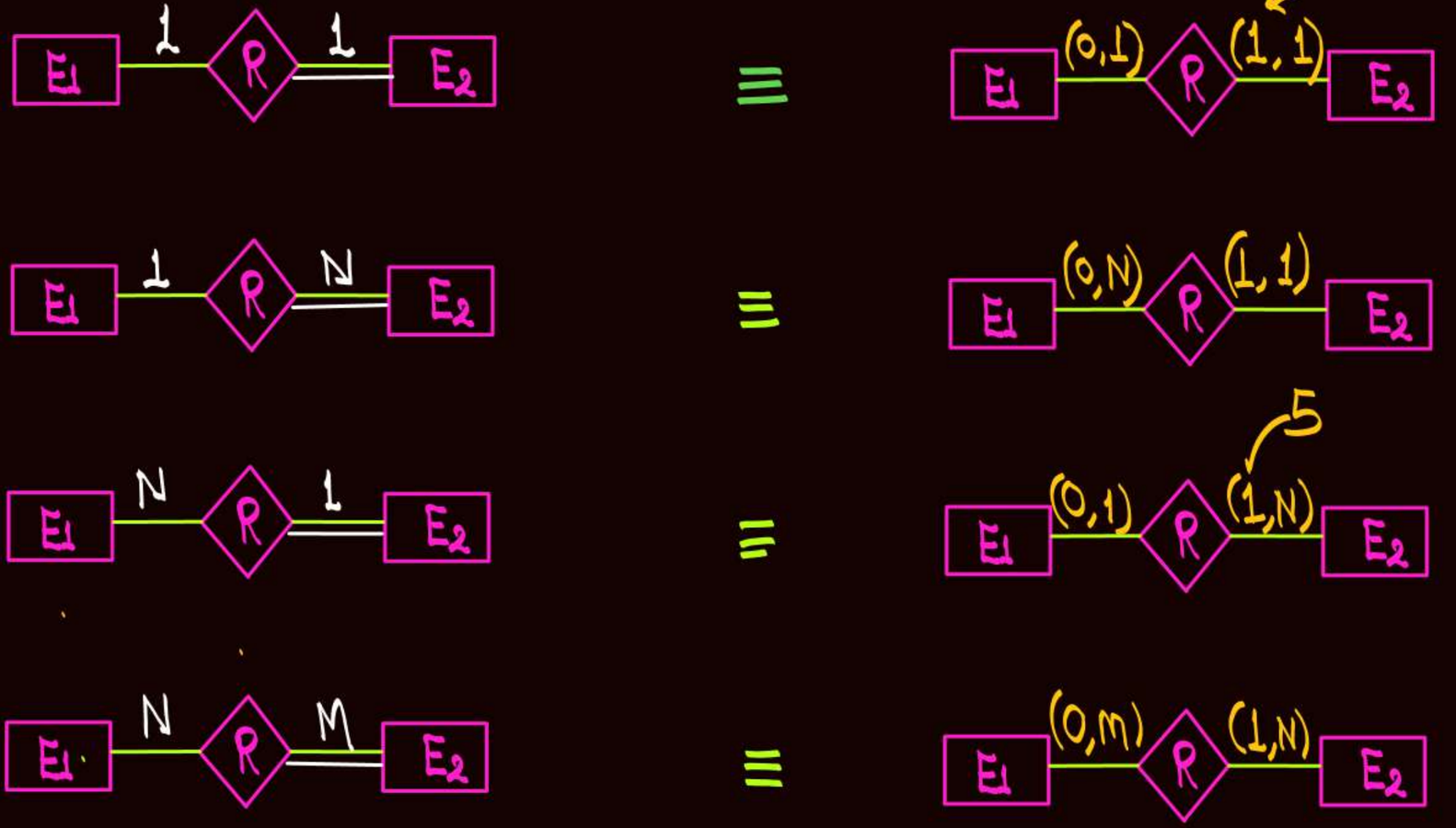


Min-Max Representation

Participation is represented using order pair of type (min, max)

1st value will be minimum value

2nd value will be maximum value



Topic : Relational Model concept

- The relational model was proposed by E.F. Codd to model data in the form of relations or tables.
- The relational model is represented by a table with columns and rows. Each row is known as a tuple.

Topic : Relational Constraints

These are the restrictions or set of rules imposed on the database content. It validates the quality of the database. It validates that various operations like data insertion, updation, etc., can be performed without affecting the integrity of the data.

Constraints could be either on a column level or a table level. The column level constraints are applied only to one column, whereas the table level constraints are applied to the whole table.

Topic : Types of Constraints on Relational Model

1. Domain constraint
2. Key constraint or Tuple Uniqueness constraint
3. Entity Integrity constraint
4. Referential Integrity constraint

Imp.

Topic : Domain constraint

Every domain must contain atomic values(smallest indivisible units) which means composite and multi-valued attributes are not allowed.

We perform a datatype check here, which means when we assign a data type to a column we limit the values that it can contain. Eg. If we assign the datatype of attribute age as int, we can't give it values other than int datatype

Topic : Key constraint or Tuple Uniqueness constraint

- These are called uniqueness constraints since it ensures that every tuple in the relation should be unique.

every relation must have a Candidate Key
↳ and for that tuples must be unique

- A relation can have multiple keys or candidate keys(minimal superkey), out of which we choose one of the keys as the primary key, we don't have any restriction on choosing the primary key out of candidate keys.

Topic : Entity Integrity constraint

- Entity integrity constraint specifies that no attribute of primary key must contain a null value in any relation.

Topic : Referential Integrity constraint

{ M. Imp }

The Referential integrity constraint is specified between two relations or tables and used to maintain the consistency among the tuples in two relations.

(in a relation) ror <- Sometimes may be on the same relation

Topic : Referential Integrity constraint

Foreign Key: A foreign key is a set of attributes in a table that refers to the primary key or alternate key of same table or some other table.

eg:

it references to the "Stu-id" attribute of Student relation

Foreign key

Primary key of student

Sid	Cid
S_1	C_1
S_1	C_2
S_3	C_2
S_4	C_3

Enroll

it is also a foreign key which will reference the course relation

Stu-id	Sname	Branch
S_1		
S_2		
S_3		
S_4		

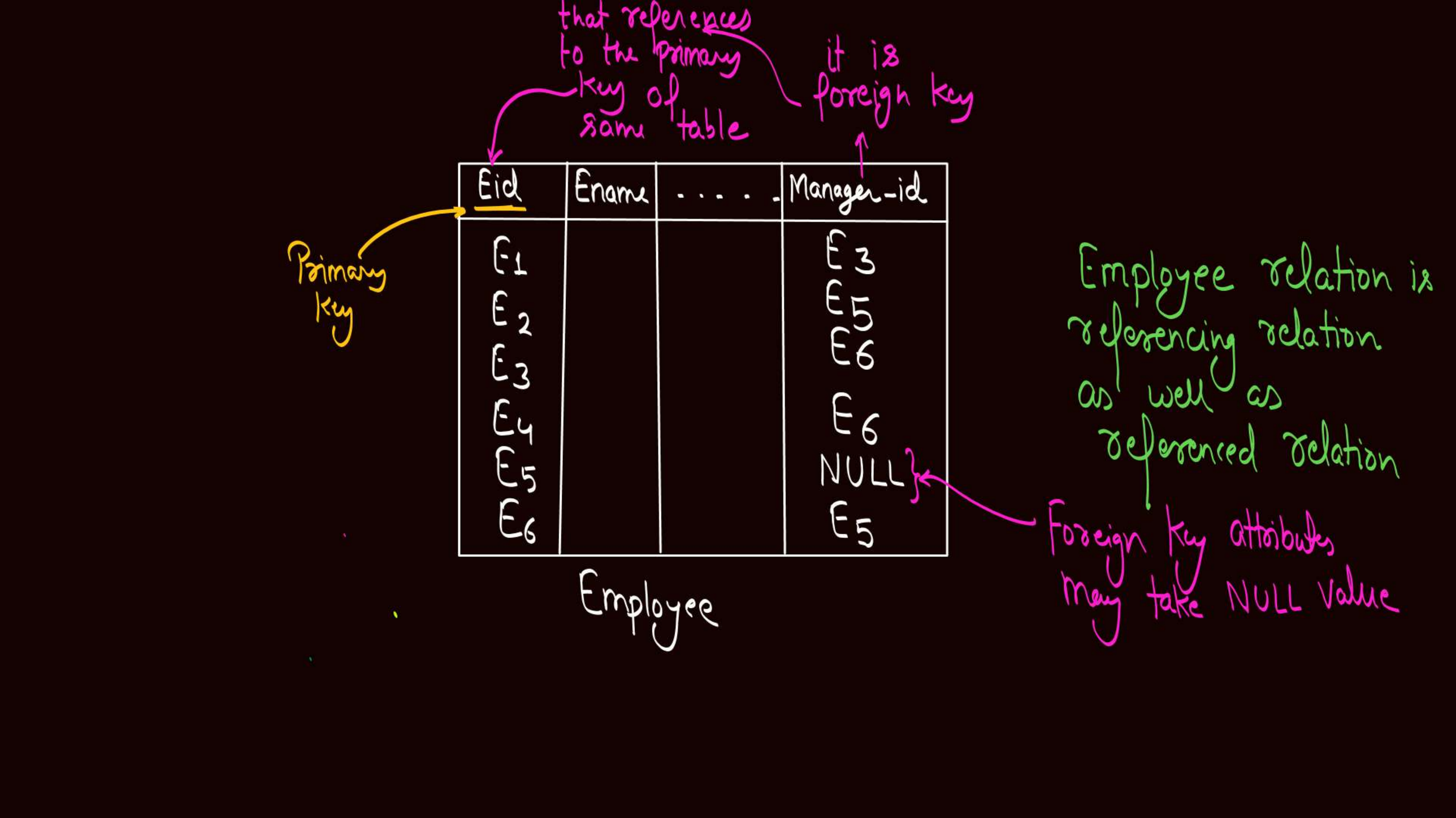
Student

Enroll relation is the "referencing" relation

Student relation is the "referenced" relation

Note: ① A relation that contains foreign key is called a referencing relation

② Relation that contains the primary key / alternate key which is being referenced by the foreign key is called a referenced relation


that references to the primary key of same table
it is foreign key
Eid
Ename
. . . . .
Manager_id
E1
E2
E3
E4
E5
E6
E3
E5
E6
E6
NULL
E5
Primary key
Employee
Employee relation is referencing relation as well as referenced relation
Foreign key attributes may take NULL value

Topic : Referential Integrity constraint

- Referential integrity constraint is enforced through a foreign key

- Let foreign key in relation R1 refers to primary key of relation R2.

  The values of the foreign key in a tuple of relation R1 can either take the values of the primary key from some tuple in relation R2, or can take NULL values, but can't be empty.

$$\{\text{Values in the Column of foreign key}\} \subseteq \{\text{Values in the Column w.r.t Primary key or Alternate key}\} \cup \{NULL\}$$

Topic : Referential Integrity constraints on Referenced relation

- **Insertion:** Insertion of a tuple in referenced relation does not result in any referential integrity violation.

- **Deletion:** Deletion of a tuple from referenced relation may result in referential integrity violation. (not always)

- **Updation:** Updation of a tuple in referenced relation may result in referential integrity violation.

Topic : Referential Integrity constraints on Referenced relation

(or Alternate key value)

- Deletion of a primary key value from the referenced relation may causes integrity violation. We may choose one of the following approaches to avoid integrity violation.

- On Delete No Action:
- On Delete Cascade:
- On Delete Set NULL:

On delete No action

If deletion of a tuple from referenced relation causes any integrity violation, then that deletion is prohibited.

{ i.e. No action will be performed }

Not even the deletion from referenced relation

On delete Cascade

If deletion of a tuple from referenced relation causes any integrity violation, then delete the tuple from the referenced relation, and also delete the corresponding tuples from referencing relation

↳ it may result in deletion of few more tuples

On delete Set NULL

Case ① If deletion of a tuple from referenced relation causes any integrity violation and Foreign key attribute is allowed to take the NULL value, then delete the tuple from the referenced relation and set the value of the Foreign key attribute as "NULL" in the tuples that causes integrity violation

---

Case ② If deletion of a tuple from referenced relation causes any integrity violation and Foreign key attribute is not allowed to take the NULL value, then deletion from the referenced relation itself is prohibited, i.e; if foreign key attribute is not allowed to take NULL values then "On delete Set NULL" is same as "On delete no action"

Topic : Referential Integrity constraints on Referenced relation

or alternate key value

- Updation in the primary key value in the referenced relation may causes integrity violation. We may choose one of the following approaches to avoid integrity violation.

- On Update No Action:
- On Update Cascade:
- On Update Set NULL:

On Update No action

If updation of a tuple in referenced relation causes any integrity violation, then that updation is prohibited.

{ i.e. No action will be performed }

Not even the updation in referenced relation

On Update Cascade

If updation of a tuple in referenced relation causes any integrity violation, then update the tuple in the referenced relation, and also update the values in the foreign key column of the referencing relation

On Update Set NULL

Case ① If updation of a tuple in referenced relation causes any integrity violation and Foreign key attribute is allowed to take the NULL value, then update the tuple in the referenced relation and set the value of the Foreign key attribute as "NULL" in the tuples that causes integrity violation

---

Case ② If updation of a tuple in referenced relation causes any integrity violation and Foreign key attribute is not allowed to take the NULL value, then updation in the referenced relation itself is prohibited, i.e; if foreign key attribute is not allowed to take NULL values then "On Update Set NULL" is same as "On Update no action"

Topic : Referential Integrity constraints on Referencing relation

Child relation

Insertion: Insertion of a tuple in referencing relation may result in referential integrity violation.

Deletion: Deletion of a tuple from referencing relation does not result in any referential integrity violation.

Updation: Updation of a tuple in referencing relation may result in referential integrity violation.

Topic : Referential Integrity constraints on Referencing relation

- If any operation in referencing relation (child relation) causes any referential integrity violation, then corresponding operation is prohibited.

2 mins Summary

Topic ER model & ER diagram

Topic Relational model & Integrity constraints

THANK - YOU