COMPUTER SCIENCE

Database Management System

ER Model

Lecture_2

Vijay Agarwal sir

TOPICS TO BE COVERED

01 Foreign Key Concepts

02 ER to RDBMS Conversions

ER MODEL

- Entity & Entity set
- Relationship & Rel^{n}ship set
- Attribute & its type
- Degree of Relationship Set
- Participation Constraint
- Cardinality Ratio

1 : 1
1 : M
M : 1
M : N

Strong Entity set
weak Entity set

& All other Concept of ER.

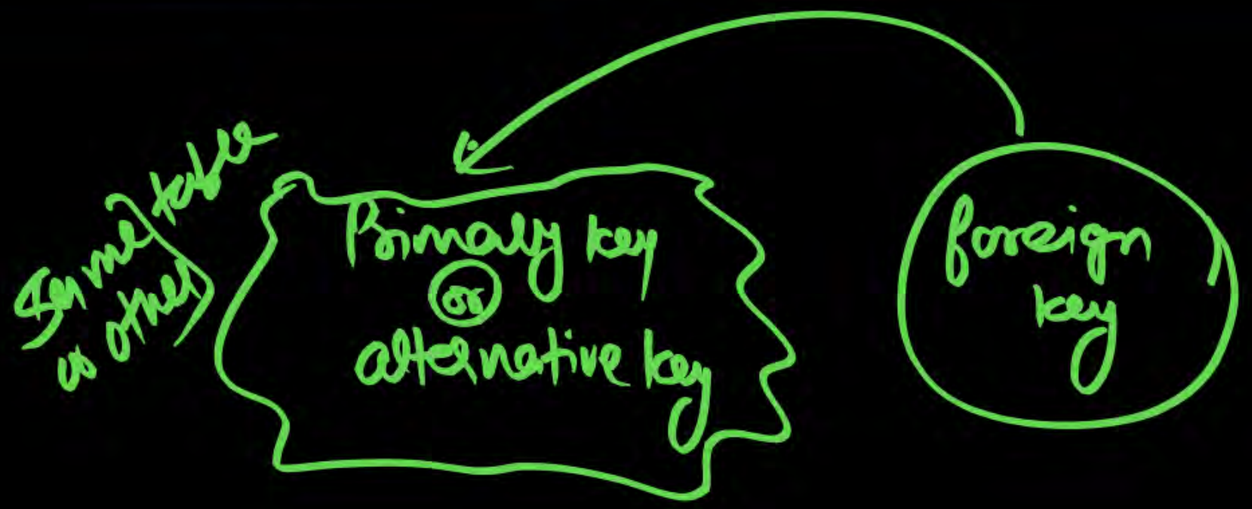
Foreign key

: is a set of attributes Reference to the Primary key (or) alternative key of same table (or) some other table.

Referential Integrity Constraints

Foreign key

Foreign key is a set of attributes that references primary key or alternative key of the same relation or other relation.



① CNC

② 1:m

foreign → NRI ⇒ Going

↳ ing

Referencing.

① Parent / Referenced Relation : The Table which is Referenced by foreign key is called Referenced / Parent Table.

② Child / Referencing Relation : The Table which Contain the foreign key is called Referencing / CHILD Table.

↓

Contain foreign key.

Referential Integrity Constraints

Foreign key

F.K

F.K

Student

Sid	Sname	Age

Referential Key must be (PK or AK)

Primary key = Sid

Referenced Relation

Enroll

Sid	Cid	Fee

Sid is foreign Key in Enroll

P.K (SidCid)

Referencing Relation

Foreign Key

Foreign Key: is a set of Attribute reference to the primary key or alterative key of the same table or same other table.

Same or Different table [Primary key or alternative key] ← Reference — [Foreign key]

It is used to relate or relation (table) with other or same relation (table)

1. **Referencing Relation:** Table which contain the foreign key is known as Referencing Relation [CHILD Relation].

2. **Referenced Relation:** Table which is referenced by foreign key is referenced relation.

Referenced Rel^n

Parent table

Referencing Rel^n

$\Downarrow$ Contain Foreign key

Foreign Key Constraint
[Referential Integrity Constraint]

F.K

STUDENT		
Sid	Sname	Login
S_1	A	-@
S_2	A	--@
S_3	B	---@
S_4	C_1	----@

F.K

Enrolled		
Sid	Cid	Fees
S_1	C_1	5K
S_1	C_2	6K
S_2	C_1	7K
S_3	C_2	8K

Sid of Enrolled table is the foreign key referencing to the primary key of student table.

[Sid: Primary Key]

Referenced Relation (Parent)

[Sid, Cid: Primary Key]

Referencing Relation (CHILD)

Note At most One Primary key Per Table (Relation)

eg) if 6 Candidate key.

→ 1 Select as Primary key

Remaining [5] C.K → Alternative / secondary key

except Primary key (5 keys)

Primary key → Unique + Not NULL

Foreign key (Sid) Reference Student

At most 1 PK Per Table.

Note

By Default Foreign key Reference to the Primary key of Referenced Relation. & at Most One Primary key Per Table So Directly Write Name of the Table.

Note Assume Loginid is the Primary key & Sid is the Alternative key in the Student Table then

Foreign key (Sid) Reference STUDENT (Sid)

(Bcz Many Alternative key are Possible)

Foreign Key Constraint
[Referential Integrity Constraint]

F.K

STUDENT		
Sid	Sname	Login
S_1	A	x-@
S_2	A	xy-@
S_3	B	xyz@
S_4	C_1	xyzw@

F.K

Enrolled		
Sid	Cid	Fees
S_1	C_1	5K
S_1	C_2	6K
S_2	C_1	7K
S_3	C_2	8K

Sid of Enrolled table is the foreign key referencing to the primary key of student table.

[Sid: Primary Key]

Referenced Relation (Parent)

[Sid, Cid: Primary Key]

Referencing Relation (CHILD)

CREATE TABLE ENROLLED

Sid Varchar (10)

Cid Varchar (10)

Fees Integer (11)

Primary key (Sid Cid)

Foreign Key (Sid) Reference Student → By Default foreign key Reference to Primary key.

When Sid is the primary key of Student ⇒ then Directly write Table Name (∵ at most 1 PK per Table)

Let login is primary key & Sid is alternative key then

Foreign Key (Sid) Reference Student (Sid) → When Sid is not primary key.

Sid is Alternative key.

(unique + NOT NULL)

P.K ← Referenced Rel^n (Parent Table)

Referencing Rel^n (CHILD Table) → Foreign key.

Note: Sometimes Same Table (Single Table) Contain P.K & F.K Both. ⇓ Different Name.

Note: The value Present in the foreign key Must be Present in Primary key of Referenced Relation & foreign key May Contain Duplicates & NULL value.

RDBMS Constraints

1. Domain Constraints (Atomic)
2. Key Constraints (Uniqueness)
3. Entity Integrity Constraint (P.K Not NULL)
4. Referential Integrity Constraint (foreign key Concept)

Q) Consider a Relation R(A B) where A is the Primary key & B is the foreign key Reference to Primary key A of the same Table. Which of the following is correct according to constraints?

OR

Which of the following Row sequence successfully inserted into R.

a) (1, 2) (2, 3) (3, 4) (4, 5) (5, 6) (6, 7)

b) (1, 2) (3, 5) (4, 6) (4, 7) (6, 7) (7, 3)

c) (1, 2) (3, 5) (4, 6) (5, NULL) (6, 5) (7, 4)

d) (1, 2) (3, 5) (4, 5) (5, 1) (NULL, 1) (2, 3)

✓ (1, 2) (2, 3) (3, 4) (4, 5) (5, 1) (6, 2)

(a) (6, 7) → '7' is Not Present in the Primary key.

(b) (4, 6) (4, 7)

A B

P.K 'A' is Not Unique

So B wrong.

(c) (1, 2) → '2' is Not Present in P.K

(8) Consider a Relation R(A B) where A is the Primary key & B is the foreign key Reference to Primary key A of the same Table. Which of the following is correct according to constraints?

(OR)

Which of the following Row sequence successfully inserted into R.

(a) (1, 2) (2, 3) (3, 4) (4, 5) (5, 6) (6, 7)

(b) (1, 2) (3, 5) (4, 6) (4, 7) (6, 7) (7, 3)

(c) (1, 2) (3, 5) (4, 6) (5, NULL) (6, 5) (7, 4)

(d) (1, 2) (3, 5) (4, 5) (5, 1) (NULL, 1) (2, 3)

✓ (e) (1, 2) (2, 3) (3, 4) (4, 5) (5, 1) (6, 2)

(d) (NULL, 1) P.K cannot be NULL.

A	B
1	2
2	3
3	4
4	5
5	1
6	2

OR

A	B
1	5
2	5
3	NULL
4	NULL
5	2
6	2

Foreign key

- Concept
- Referencing & Referenced Rel^n.
  - F.K

The value Present in F.K Must be Present in Primary key of Referenced Rel^n & F.K May Contain Duplicate & Null value.

Foreign Key Constraint
[Referential Integrity Constraint]

P →

STUDENT		
Roll No	Name	Branch
1	A	CSE
2	B	IT
3	C	CSE
4	D	IT

Referenced Relation

(Parent)

← F.K

Registration		
CNo	Cname	Roll No
101	DBMS	1
102	OS	1
103	CD	3
104	TOC	NULL.
105	C++	6

→ Not Present in P.K

Referencing Relation

(CHILD)

Referenced Rel^n
(Parent Table)

↓

✓ Insert

✗ Delete

Referencing Rel^n
(CHILD Table)

↓

✓ Delete

✗ Insert

Note: Deletion from Referenced Relation & Insertion into Referencing Relation May Violate Referential Integrity Constraint.
(Foreign key Constraint)

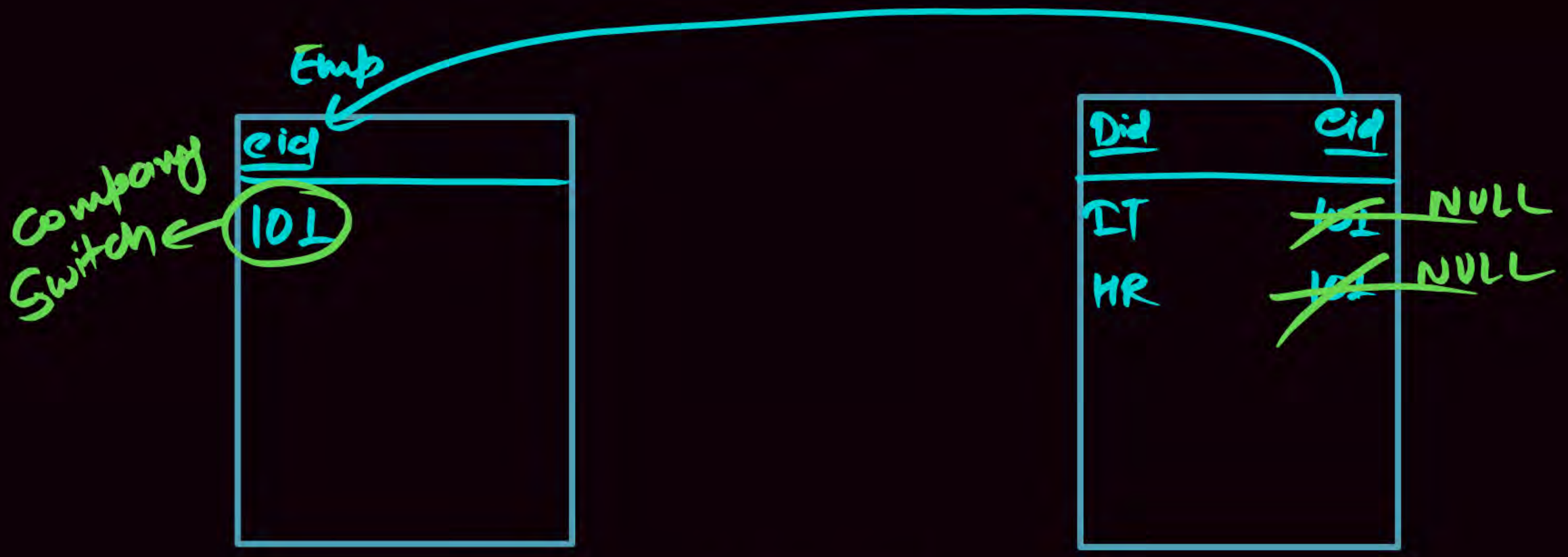
Note: The value present in Foreign key must be Present in Primary key of Referenced relation

Foreign key may contain duplicate & NULL values.

(Parent table)	CHILD table
Referenced table	Referencing Relation
✓ Insert < 4 D ECE>	✗ Insert < 105 DSA 67
✗ Delete < 1 A CSE>	✓ Delete < 103 CD 3>

Note: Deletion from the Referenced Relation and Insertion into Referencing Relation may violate Foreign key constraint.

Note: A Relation can Act as Parent & CHILD i.e. Relation may contain a primary key & a Foreign key that Refer to the same Relation.



ON DELETE SET NULL

Foreign Key Constraint
[Referential Integrity Constraint]

ON DELETE CASCADE

STUDENT		
P → Roll No	Name	Branch
1	A	CSE
2	B	IT
3	C	CSE

Registration		
CNo	Cname	Roll No ← F.K
101	DBMS	1
102	OS	1
103	CD	3
104	TOC	NULL

ON DELETE CASCADE

Referenced Relation (Parent)

Referencing Relation (CHILD)

Referential Integrity Constraint

(1) Referenced Relation

(i) Insertion : No Violation

(ii) Deletion : May cause Violation if Primary key is used by referencing relation

I. ON DELETE NO ACTION.

V.V.Imp. II. ON DELETE CASCADE.

III. ON DELETE SET NULL.

I

Foreign key (Sid) Reference Student(Sid)
ON DELETE NO ACTION.

Foreign key (Sid) Reference Student(Sid)
ON DELETE CASCADE.

Foreign key (Sid) Reference Student (Sid)
ON DELETE SET NULL

Foreign key

Referenced Relation

1. Insertion : No violation
2. Deletion : [May cause violation]

(a) On delete no action : Means if it cause problem on delete then deletion is not allowed on table.

(b) On delete cascade : If we want to delete primary key value from referenced table then it will delete that value from referencing table also.

(c) On delete set null : If we want to delete primary key value from referenced table then it will try to set the null values in place of that value in referencing table.

ON DELETE CASCADE: Whenever Primary key Deleted then Corresponding tuples (that P.K used as F.K) from the Referencing Relation also Deleted Cascadelly.

Note: Its [ON Delete Cascade] Not Suggestable when One Table Contain Primary key & foreign key Both

eg: Like in Previous (eg) Deletion of One Tuple (E2 NULL) On Delete Cascade Complete Table Deleted.

Foreign key Supervision id Reference Empleid
ON DELETE CASCADE

Q. If we wants to Delete (E2, NULL) then How many tables additionally Deleted to Preserve Referential Integrity ?

ON DELETE CASCADE.

(E2, NULL) ⇒ P.K E2 Delete, then Due to P.K E2 ⇒ (E1 E2) (E3 E2) also Deleted

P.K Employee F.K

eid	ename	Dno	Supervision id
E1	A	x	E2
E2	B	y	NULL
E3	C	z	E2
E4	D	x	E3
E5	A	y	E4
E6	E	z	E5

Foreign key supervision id

Foreign key Supervision id Reference Empleid
ON DELETE CASCADE

(E1 E2) (E3 E2) Now Primary key E1 & E3 also Deleted

Due to 'E3' ⇒ (E4 E3) Deleted

Due (E4, E3) ⇒ P.K 'E4' ⇒ (E5, E4) also Deleted

Due to (E5, E4) P.K 'E5' ⇒ (E6 E5) Deleted

for this example
So Complete Table Deleted.

Employee

eid (P.K)	ename	Dno	Supervision id (F.K)
① E1	A	x	E2
E2	B	y	NULL
② E3	C	z	E2
③ E4	D	x	E3
④ E5	A	y	E4
⑤ E6	E	z	E5

Foreign key supervision id

NOTE:

If foreign key field is not null attribute then “On delete set null” is same as “on delete no action.”

Foreign key

3. Updation : [May cause violation]
   (a) On update no action
   (b) On update cascade
   (c) On update set null

Referencing Relation (CHILD)

1. Insertion : [May cause violation]
2. Deletion : No violation
3. Updation : [May cause violation]

NOTE:

If integrity violation occurs because of insertion or updation in referencing table then restrict insertion and updation.

Example

P.K F.K

A	B
2	4
3	4
4	5
5	4
6	2

B is foreign key

Referencing A,

Delete (2, 4) and

on delete cascade

A	B
3	4
4	5
5	4

Result

So, If we delete (2, 4) then PK "2". gets deleted from the table and all the tuples in which B is referencing PK.2" also gets deleted.

Q.1

The following table has two attributes A and C where A is the primary key and C is the foreign key referencing A with on-delete cascade.

The set of all tuples that must be additionally deleted to preserve referential integrity when the tuple (2, 4) is deleted is:

(A) (,3 4) and (6, 4)

(B) (5, 2) and (7, 2)

(C) (5, 2), (7, 2) and (9, 5)

(D) 1

Ans [C].

A	C
2	4
3	4
4	3
5	2
7	2
9	5
6	4

(5,2) (7,2)

(9,5)

Referenced Referring

Q2 Let R (a, b, c) and s(d, e, f) be two relation in which d is the foreign key of S that refers to the primary key of R. Consider the following four operations on R and S.

(i) Insert into R

(ii) Insert into S (Referring)

(iii) Delete from R

(iv) Delete from S

Which of the following is true about the referential integrity constraint above?

A. None of (i), (ii), (iii), or (iv) can cause its violation

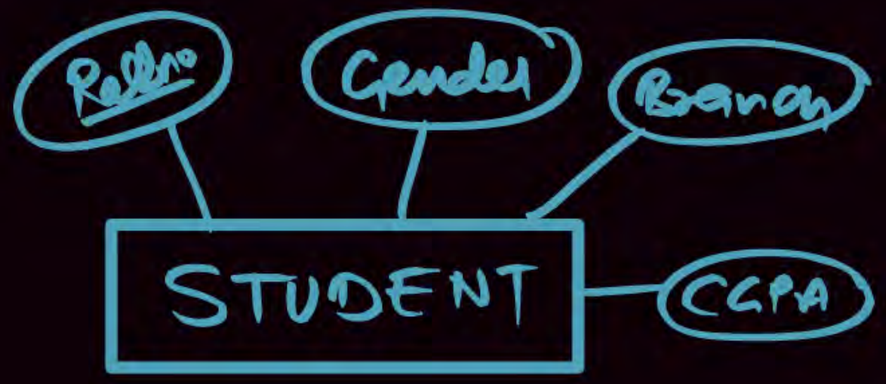
B. All of (i), (ii), (iii), and (iv) can cause its violation

C. Both (i) and (iv) can cause its violation

D. Both (ii) and (iii) can cause its violation

ER MODEL

1



ER		RDBMS
Entity Set	→	Relation(Table)
key Attribute	→	Primary key

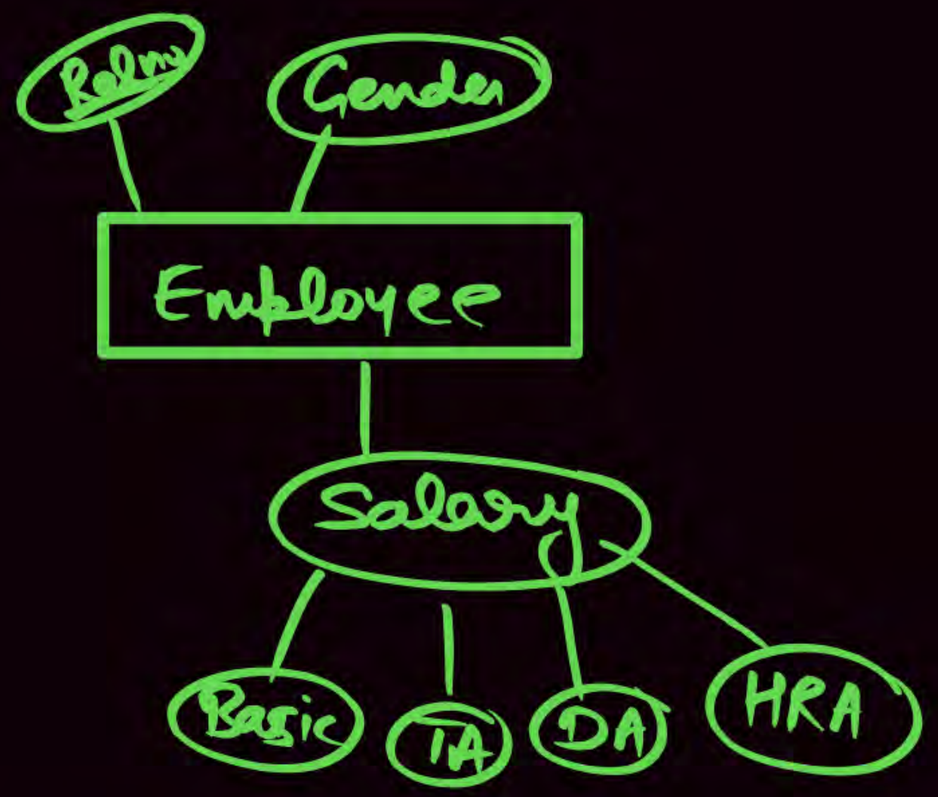
RDBMS

STUDENT

RollNo	Gender	Branch	CGPA

ER MODEL

(2)



Composite Attribute → Set of Simple Component

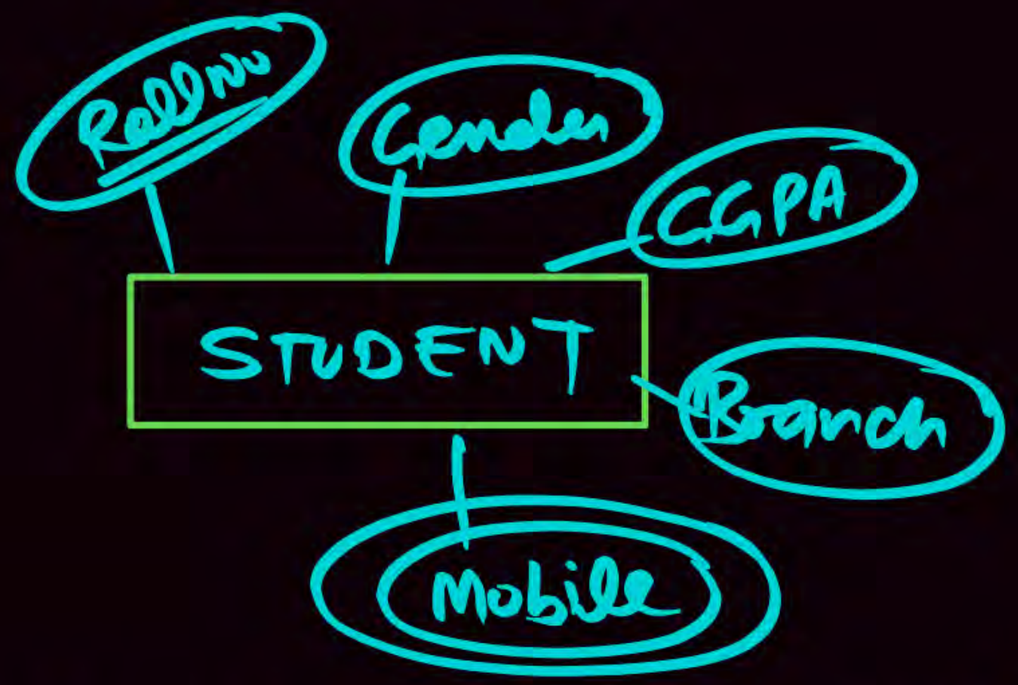
RDBMS

~~Salary~~

Employee.

RollNo	Gender	Basic	TA	DA	HRA

ER MODEL



RDBMS

STUDENT

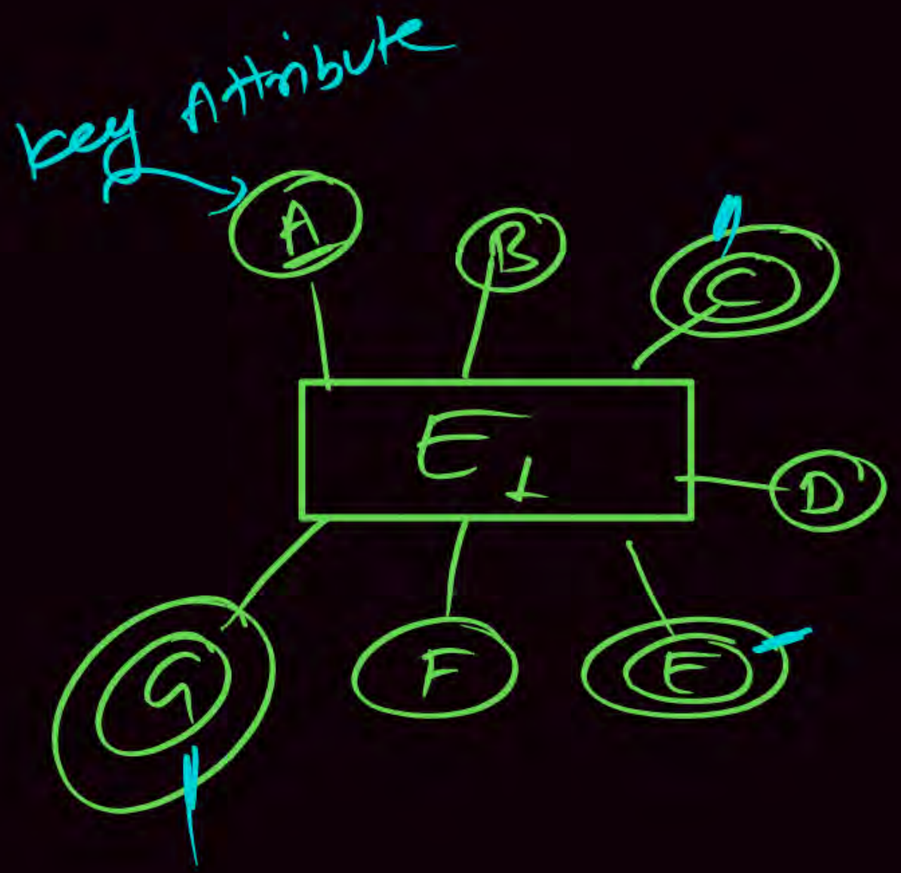
RollNo	Gender	CGPA	Branch

STUDENT'

Rollno	Mobile

If ER MODEL Contain Multivalued Attribute then In RDBMS 2 Table Required

- I^{st} key + ALL other Attribute
- IInd key + ALL Multivalued Attribute



(a) 2
(b) 3
(c) 1
(d) None

E1

A	B	D	F

E1'

A	C	E	G

2 Table Required

Q. The term in list A have been mapped to list B so that is corresponds to the mapping process of ER MODEL into relational. Which of the following represent the mapping process? **[MCQ]**

List-A	List-B
A. Entity type → Relation	1. Primary key (or alternate key)
B. Key attributes → P.K	2. Child table
C. Composite attribute →	3. Set of simple component attributes
D. Multivalued attribute → Child Table	4. Relation

A-4
B-1
C-3
D-2

(A) A-3, B-1, C-4, D-2

(B) A-4, B-1, C-3, D-2 ✓

(C) A-3, B-2, C-2, D-4

(D) A-4, B-1, C-2, D-3

Mapping [Cardinality constraints of relationship set]

(For binary relationship)

Partial participation on both side of binary relationship

- One to Many : Merge relationship set towards many side. So, 2 relational tables.
- Many to one : Merge relationship set towards many side. So, 2 relational tables.
- One to one : Merge relationship set any one side. So, 2 relational tables.
- Many to Many : Separate table for each entity set and relationship set. so, 3 relational tables.

Mapping [Cardinality constraints of relationship set]

(For binary relationship)

Full participation on "one" side of many to one relationship

Merge the entities and relationship set into single relational table. So, 1 table.

Mapping [Cardinality constraints of relationship set]

(For binary relationship)

Full participation on "Many" side of Many-to-one relationship

Merge relationship set towards many side. So, 2 relational tables.

Mapping [Cardinality constraints of relationship set]

(For binary relationship)

Full participation on any "one" side in one-to-one relationship

Merge the entity sets and relationship set into single table. So, 1 table.

Mapping [Cardinality constraints of relationship set]

(For binary relationship)

Full participation on any "Many" side of Many-to-Many relationship

Merge relationship set towards any "Many" side of relationship. So, 2 table.

Mapping [Cardinality constraints of relationship set]

(For binary relationship)

Full participation on both side of relationship

1 : 1

1 : M

M : 1

M : N

Merge the entity sets and Relationship into single Relational table so, 1 relational table.

Self-Referential Relationship Set

(Recursive entity set)

Entities of entity set (E) related to some other entity of same entity set (E).

Recursive
Entity set

Self-referential relationship set

- Binary relationship set
- 1: M, M:1, 1:1, M:N can be any of the relationship

Q. Consider the following entity relationship diagram(ERD), where two entities E1 and E2 have a relation R of cardinality 1:m

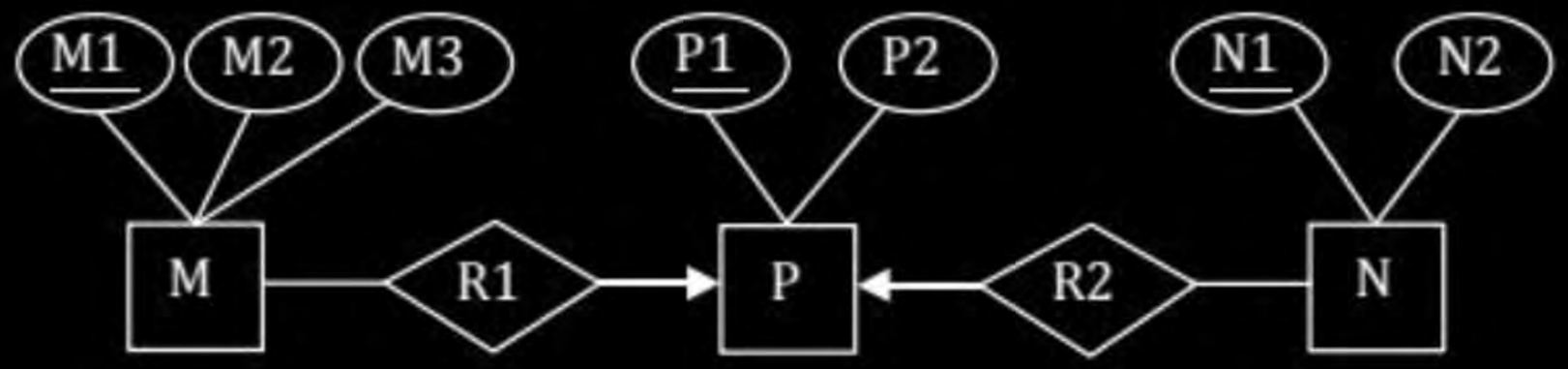
The attributes of E1 are A11, A12 and A13 where A11 is the key attribute. The attributes of E2 are A21, A22, A23 where A21 is the key attribute and A23 is a multi-valued attribute. Relation R does not have any attribute. A relational database containing minimum number of tables with each tables satisfying the requirements of the third normal from (3NF) is designed form the above ERD. The number of tables in the database is [GATE-2004 : 2 Marks]

(A) 2 (B) 3 (C) 5 (D) 4

Q.

Common Data for Question

Consider the following ER Diagram



(i) The minimum number of tables needed to represent M, N P, R1, R2 is [GATE-2008 : 2 Marks]

A 2

B 3

C 4

D 5

(ii) Which of the following is a correct attribute set for one of the table for the correct answer to the above question?

GATE-2008 : 2 Marks]

A {M1, M2, M3, P1}

B {M1, P1, N1, N2}

C {M1, P1, N1}

D {M1, P1}

E-R Model:

Entity Relationship Diagrams used to represent Diagrammatic design [High Level Design] of Databases.

DB Design Steps:

1. Requirement: What type of data stored and what operation required etc.
2. Conceptual and Logical: [ER Diagram]
3. ER Diagram to RDBMS table design and apply normalization.
4. Physical DB Design (Indexing Design)
5. User interface Design && Security Design

High Level Design

Low Level Design

Main Components in ERD:

1. Attributes
2. Entity Sets
3. Relationship Sets

Attribute

Attributes:

Key Attributes:

Multi valued Attribute:

Derived Attribute:

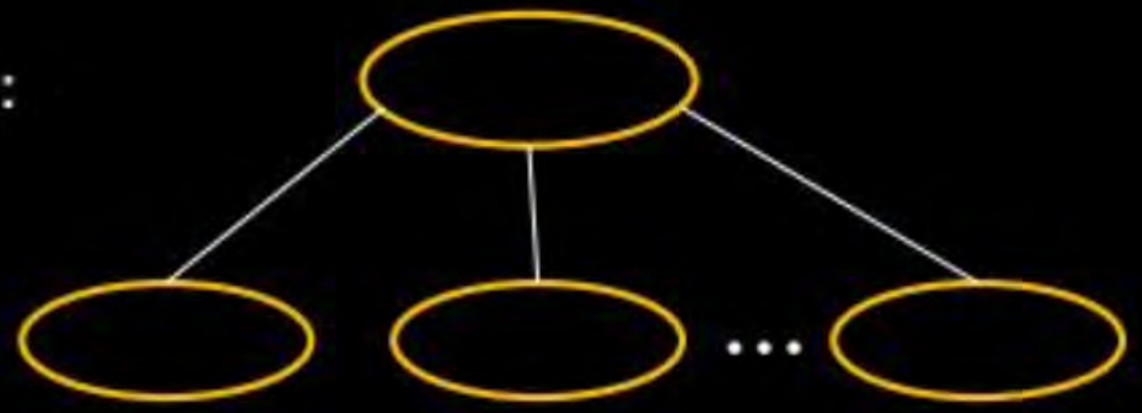
(Value derived from Other stored value)

Example:

DOB → Age ⇒ System Date - DOB ≡ age

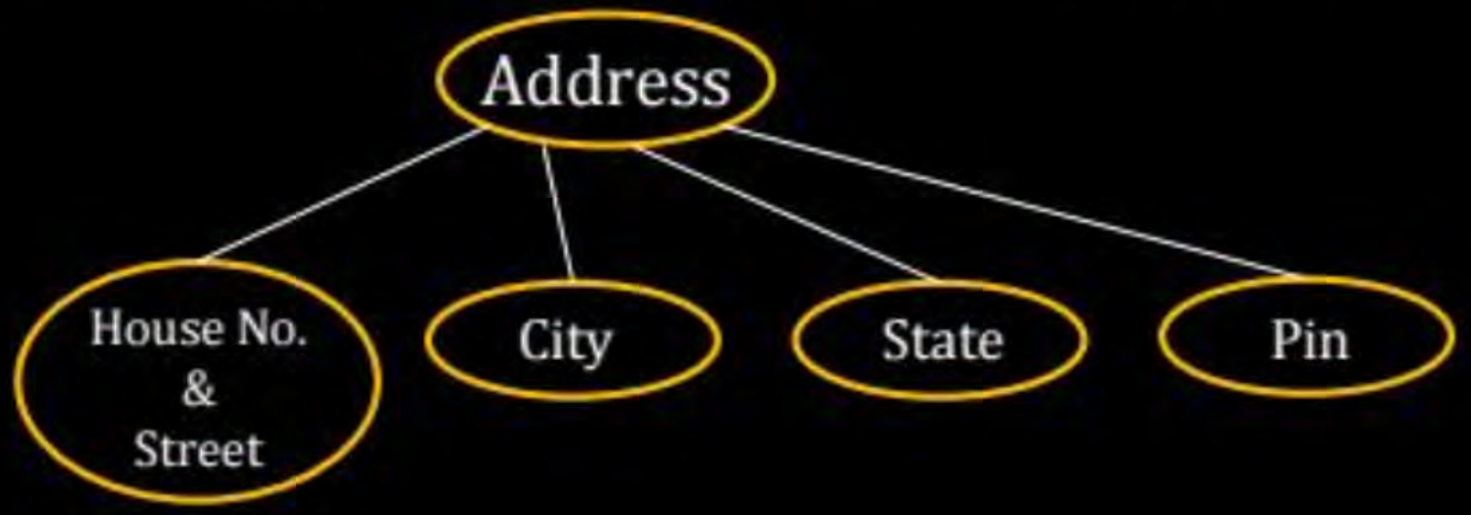
Attribute

Composite Attribute:



Attribute Which can represent as two or more attributes

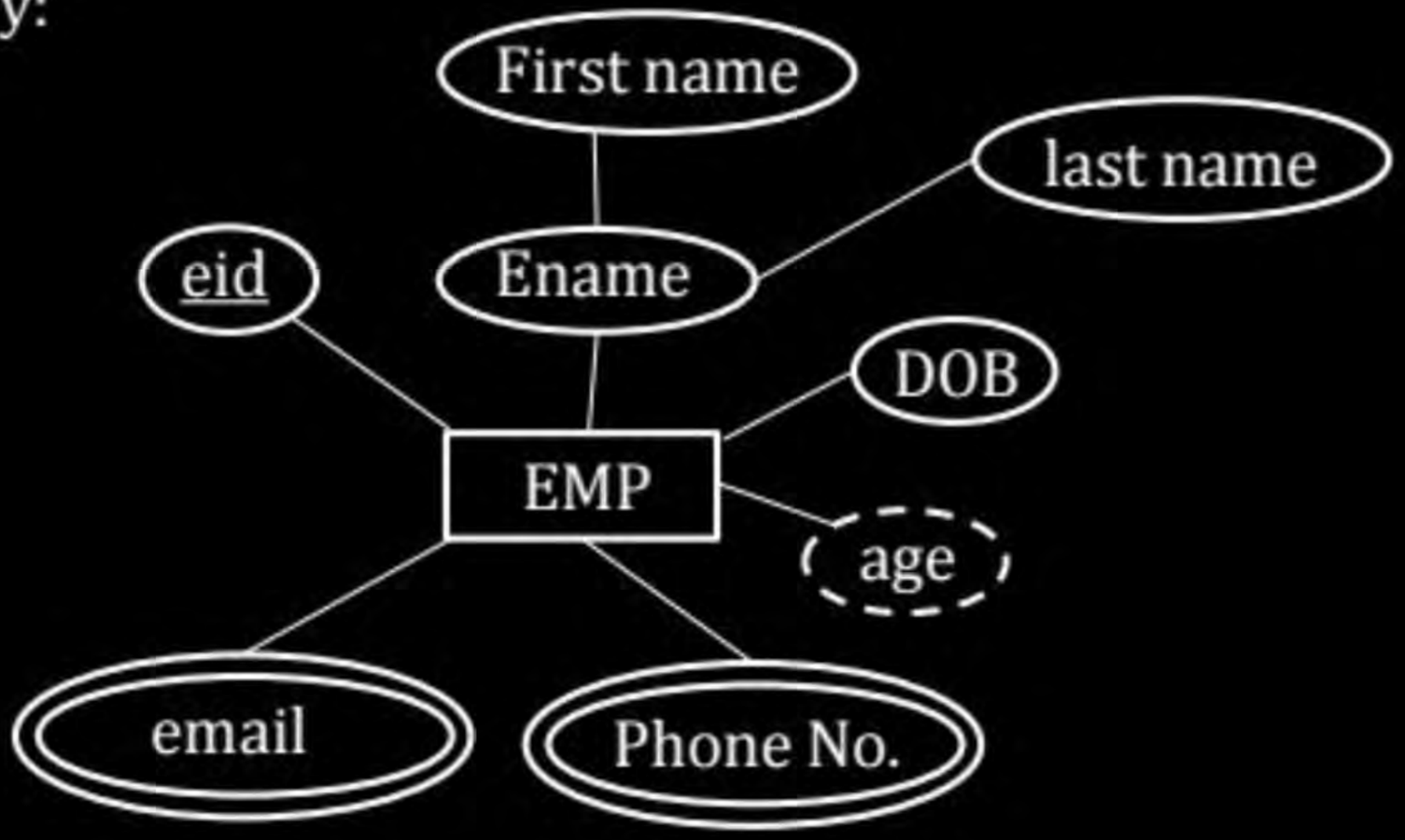
Example:



Entity Set:

Set of similar entities (recorded/objects).

Represented by:



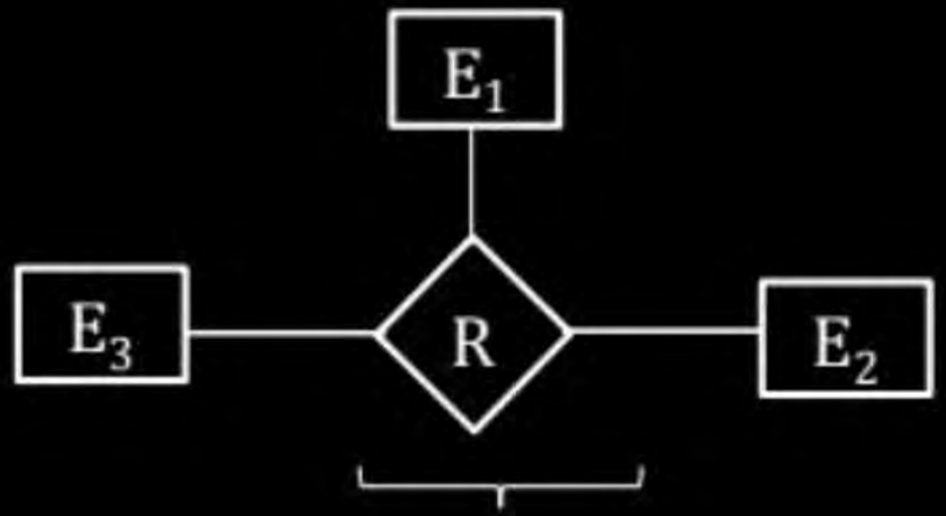
Relationship Set:

Used to relate two or more entity set.

Represented by:

1.

Binary Relationship

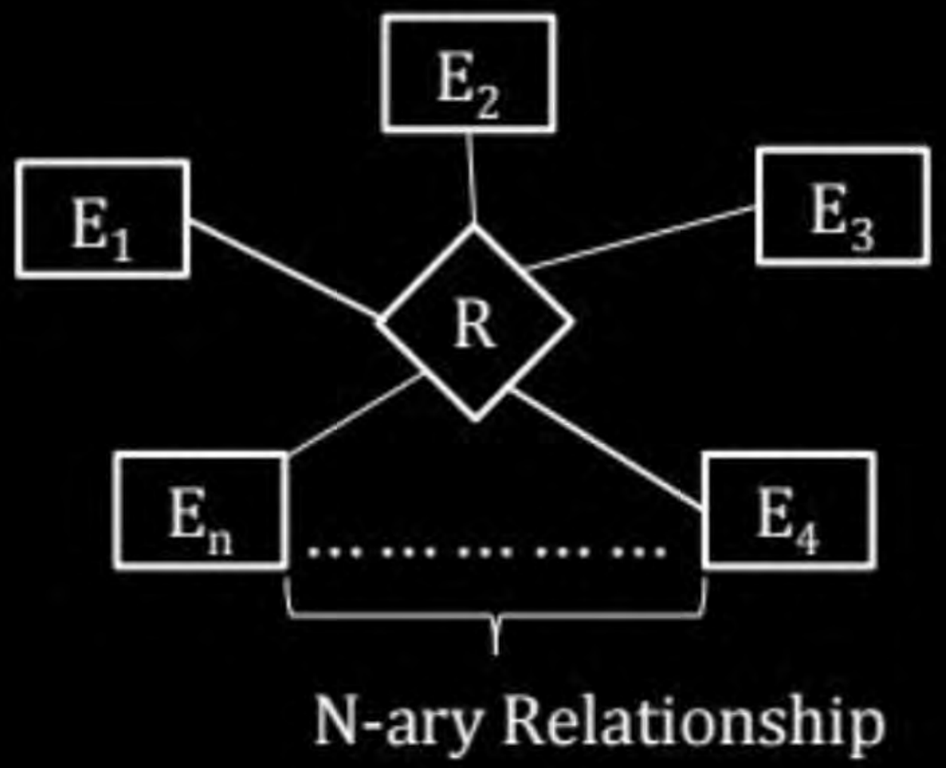
2. 

Ternary Relationship

Relationship Set:

Used to relate two or more entity set.

Represented by:

3. 

Participation:

If every entity of entity set (E1) must participate with relationship set than Total Participation.

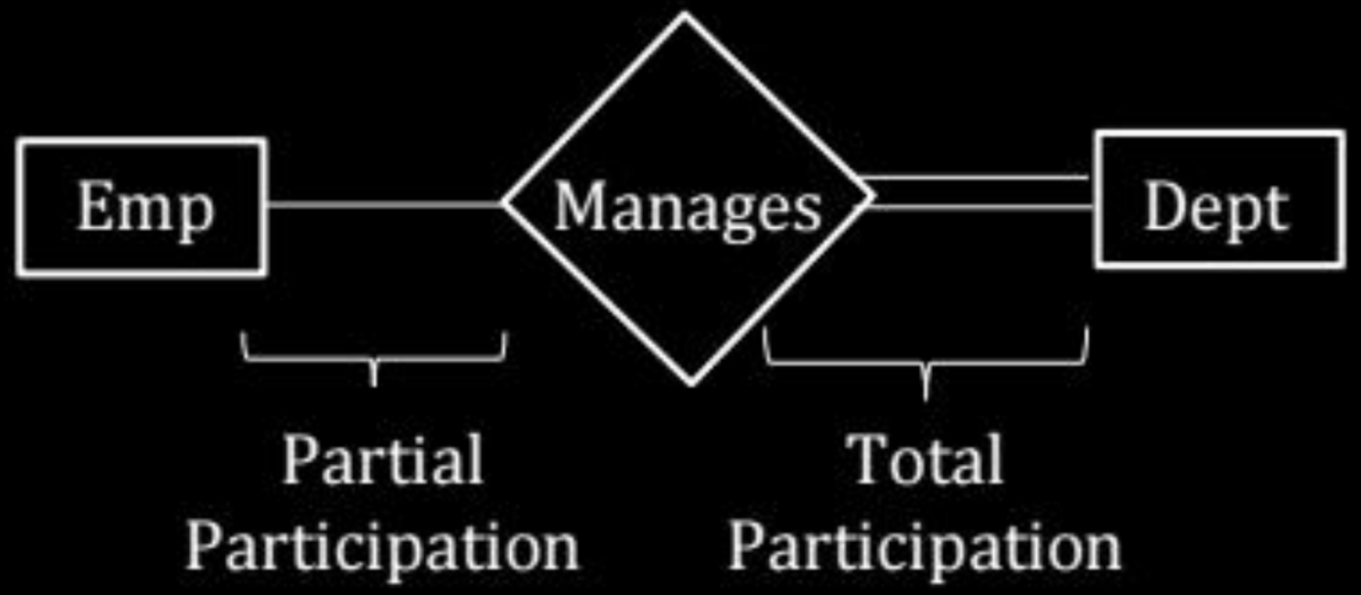
[must be 100% participation]

Otherwise:

Partial Participation:

May or may not 100% participation

Example: "Emp" & "Dept" Entity Set.
"Manages" Relationship Set such that each dept there must be a manager.



Mapping:

Mapping [cardinality of Relationship set]:

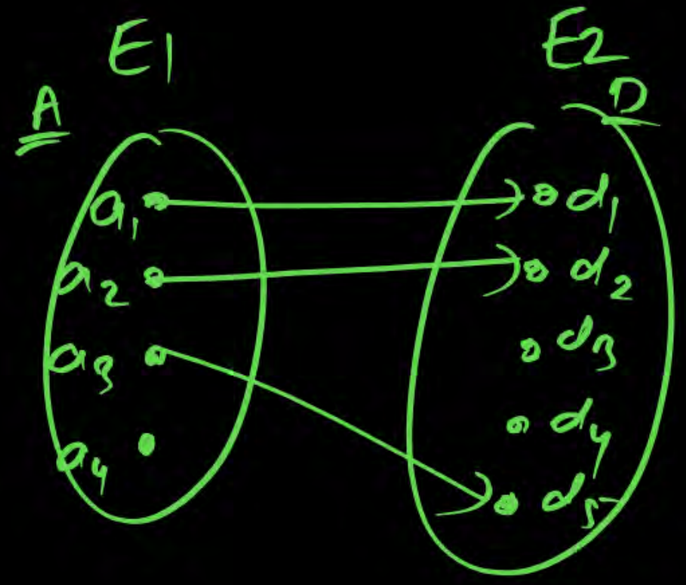
One: [At most one] [0.....1]
Many: [0 or more] [0.....M]

Possible Mapping of binary relationship sets are:

1) One : One
2) One : Many
3) Many : One
4) Many : Many

Candidate key's of Relationship set is Based on mapping

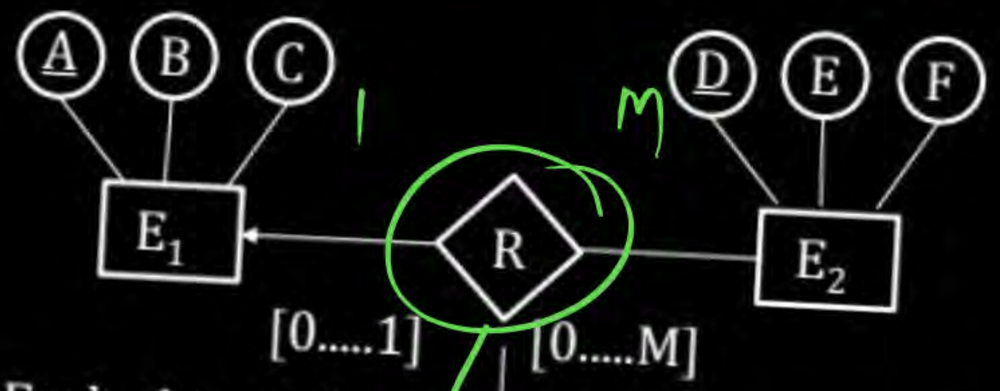
One to One:



R (A D) ≡ {A, D}

a_1	d_1	1 : 1 mapping
a_2	d_2	candidate key of
a_3	d_5	Relationship set

One: Many:



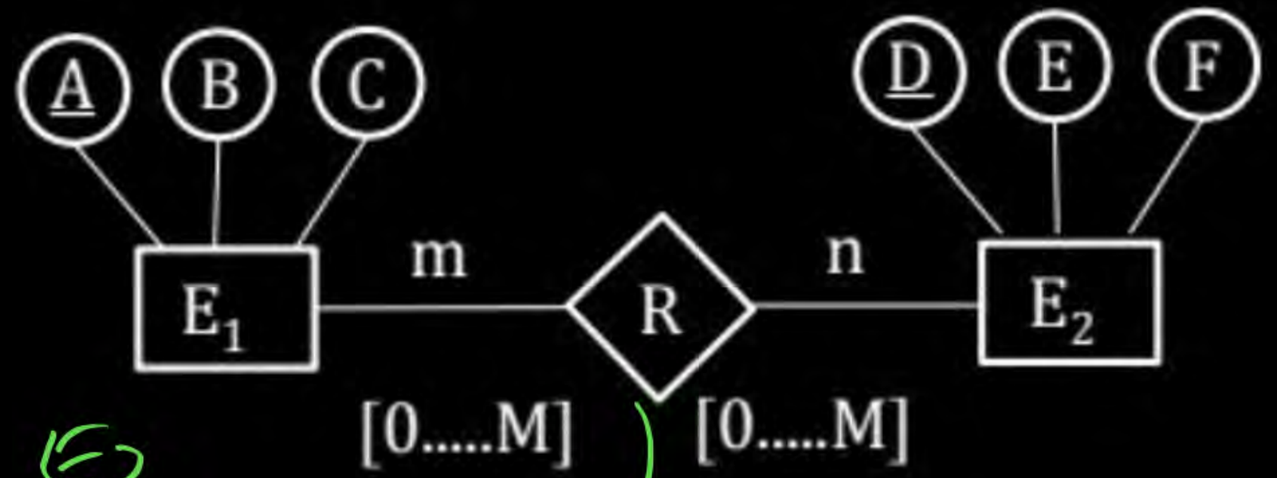
Each object of E_2 allowed to pair by at most one entity of E_1

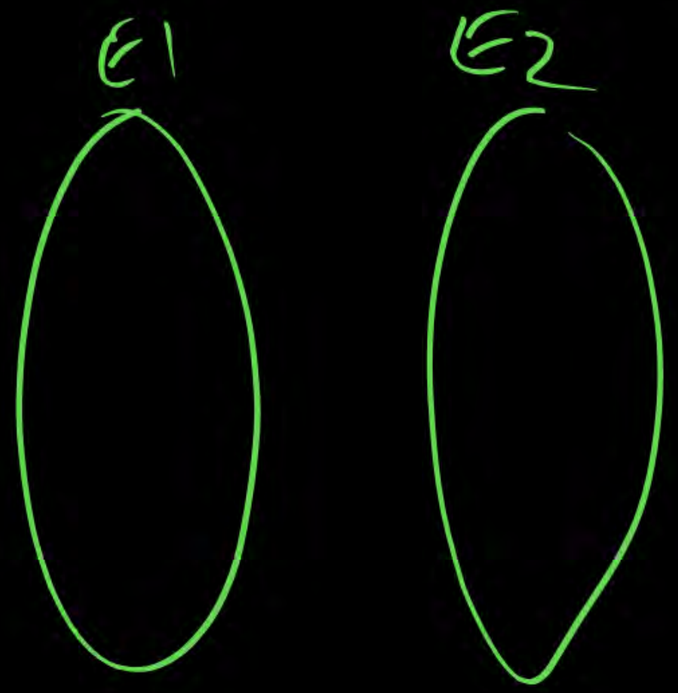
Each entity of E_1 can pair with many E_2 entity.

Candidate key of R (A D) ≡ { D }

a_1	d_1	: for 1: M
a_1	d_2	Relationship
a_2	d_3	

Many to Many:



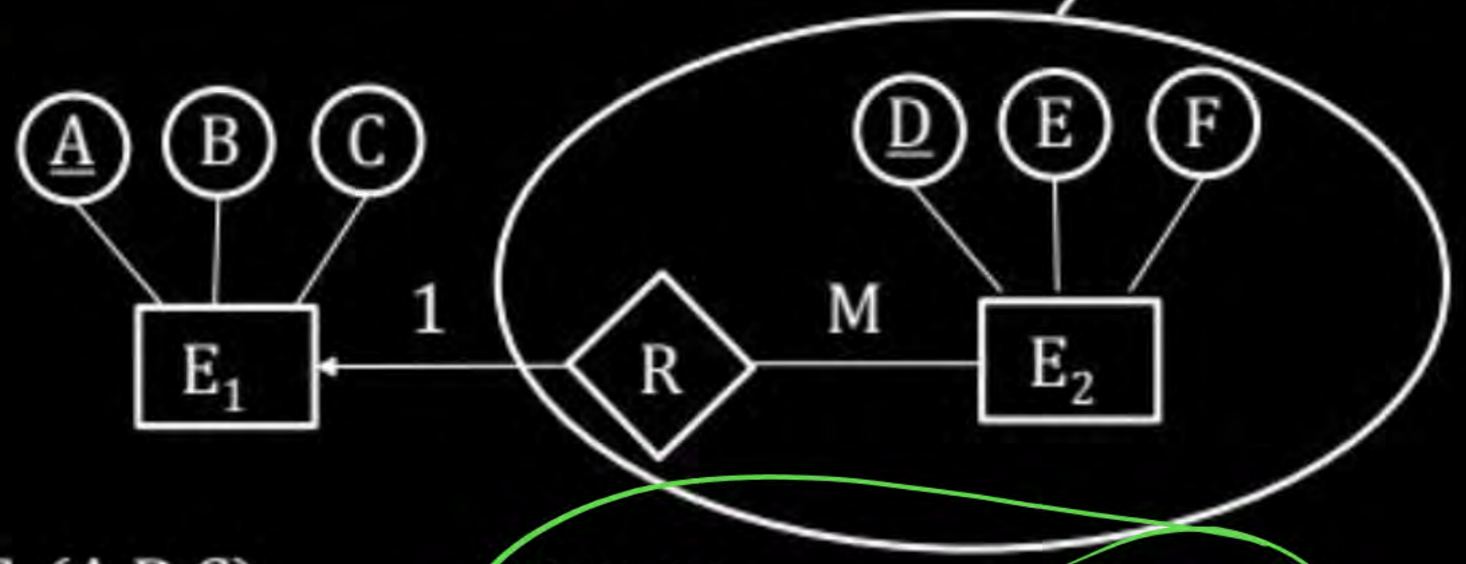


R (A D) ≡ {AD}

a_1	d_1	M:N ⇒
a_1	d_2	for many to many mapping
a_2	d_2	Candidate Key for
a_3	d_3	Relationship Set is
		{AD}

RDBMS Design of Given ERD:

1 : M Relationship:



$E_1(\underline{A}\ B\ C)$	$R(A\ \underline{D})$	$E_2(\underline{D}\ E\ F)$
a_1 – –	$a_1\ d_1$	d_1 – –
a_2 – –	$a_1\ d_2$	d_2 – –
a_3 – –	$a_2\ d_3$	d_3 – –
a_4 – –		d_4 – –
$A \rightarrow BC$	$D \rightarrow A$	$D \rightarrow EF$
$\{\underline{A}\}$	$\{\underline{D}\}$	$\{\underline{D}\}$

$\{A \rightarrow BC\}$ $E_1(\underline{A}\ B\ C)$ $E_2R(\underline{D}\ E\ F\ A)$ $\{D \rightarrow A\ E\ F\}$

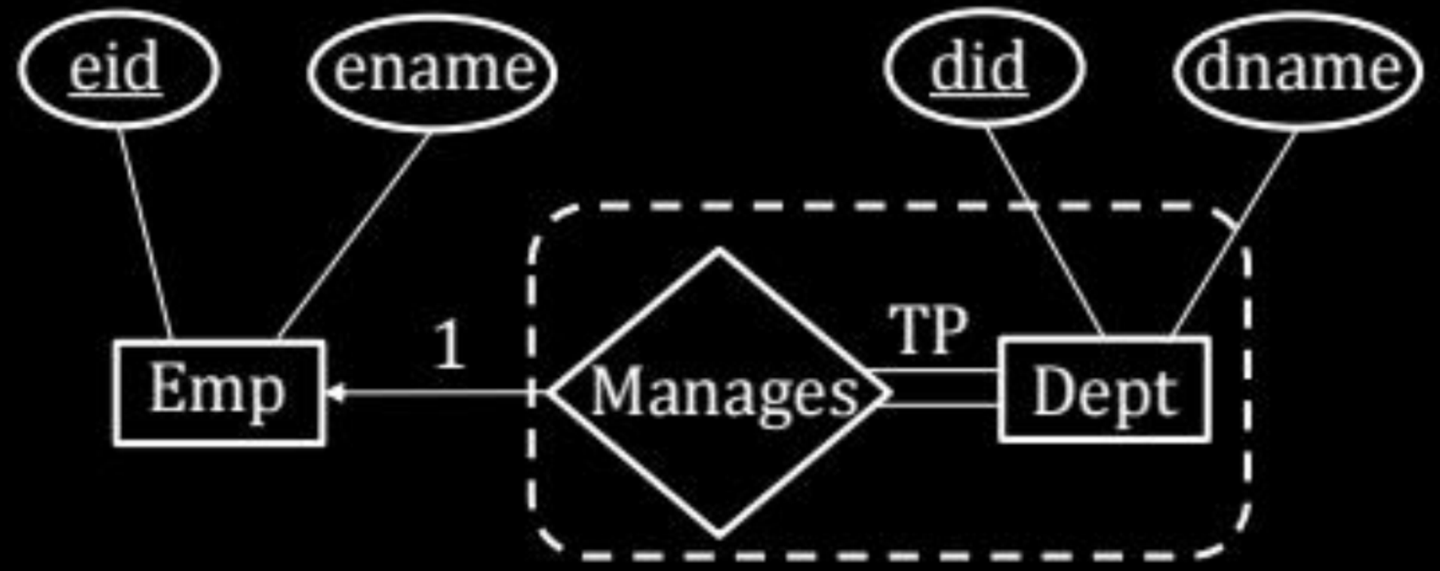
a_1 – –	d_1 – – a_1
a_2 – –	d_2 – – a_1
a_3 – –	d_3 – – a_2
a_4 – –	d_4 – – NULL

Partial participation between E_2 and R

Due to partial participations.

$\therefore$ minimum 2 relational Table Required and 1 Foreign Key required for given ERD

Given ERD
Example:



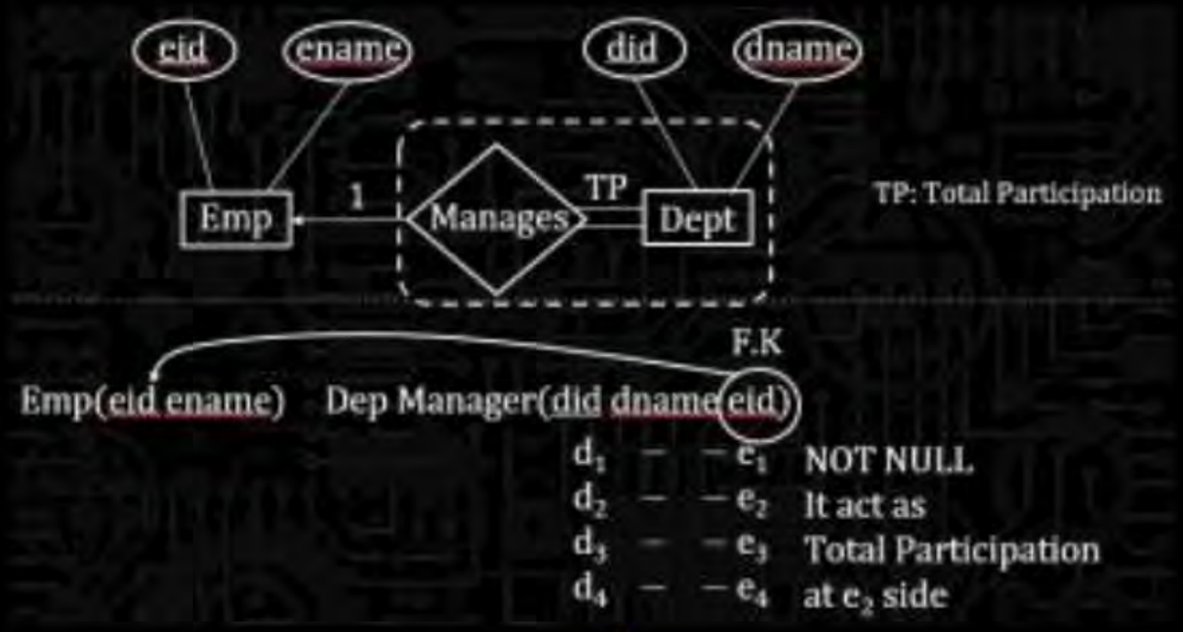
TP: Total Participation

F.K

Emp(eid ename) Dep Manager(did dname eid)

d_1	–	–	e_1	NOT NULL
d_2	–	–	e_2	It act as
d_3	–	–	e_3	Total Participation
d_4	–	–	e_4	at e_2 side

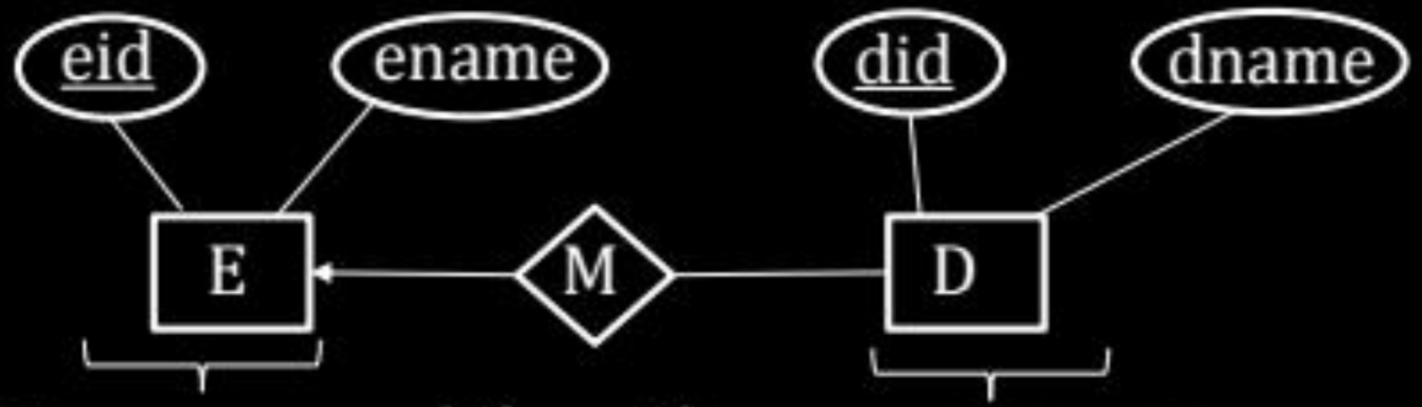
Given ERD



Emp_Dept_Manages
(did dname eid ename)
did → dname
did → eid
(_____ e_5 A)
Not allowed because e_5 not in relationship

```
CREATE TABLE Dept_manages
(did varchar(10) primary key,
dname varchar(30),
eid varchar(10) NOT NULL,
FOREIGN KEY(eid)
REFERENCES Emp);
```

If 1 : M relationship set merges into left side Entity set:



eid → ename did → eid did → dname

E_M(eid ename did) D(did dname)

eid	ename	did
e_1	A	d_1
e_1	A	d_2
e_1	B	d_2
e_1	C	d_3
e_1	D	Null

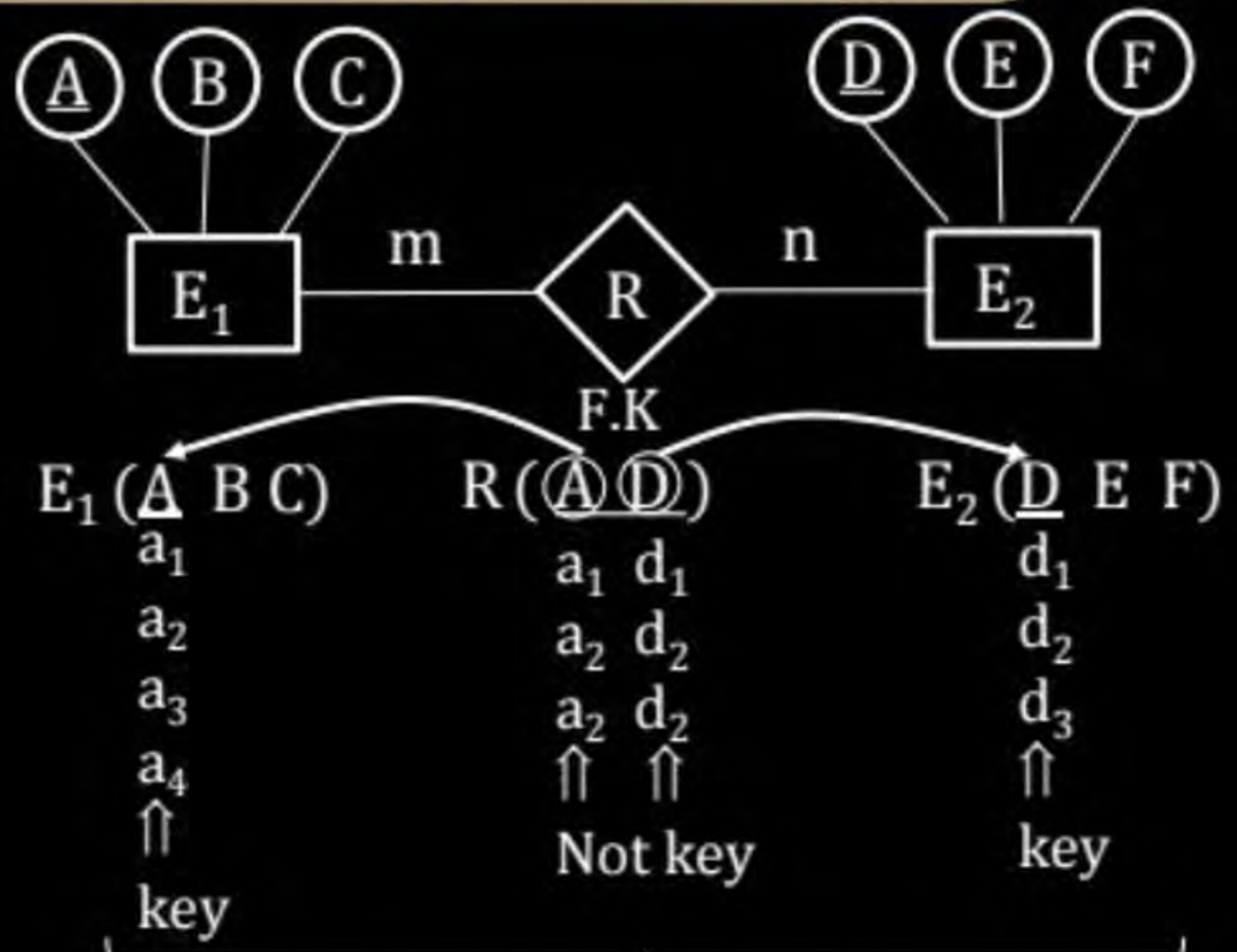
Not valid as did is key of E_M
(Partial Participation)
[lost at E side]

Disadvantage:

1) Data Redundancy Occurs
   eid $\rightarrow$ ename
   Not Super Key(S.K)

2) Partial Participation will be lost
   [Not possible to insert employees who are not manager of dept.]

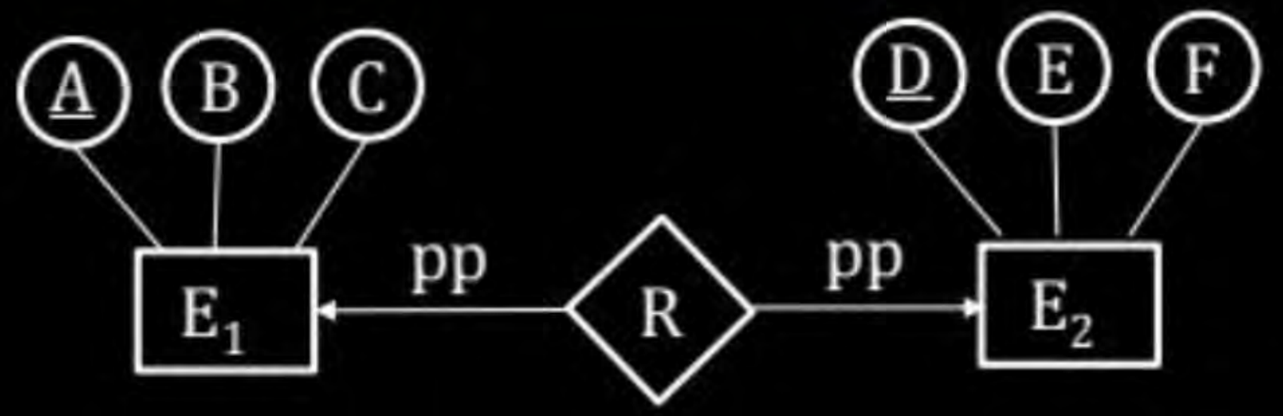
Many : Many Relationship:



Minimum 3 relational table required and 2 foreign key required.

M:N relationship set not allowed to merge with any entity set.

One : One Relationship:



E_1, E_2 entity sets & R relationship set related between E_1 and E_2 with 1:1 cardinality and partial participation(pp) at both side.

E_1 (A B C)	R (A D)	E_2 (D E F)
a_1	a_1 d_2	d_1
a_2	a_2 d_3	d_2
a_3	a_3 d_4	d_3
a_4	⇑ ⇑	d_4
⇑	key key	⇑
key	[A→D	key
[A→BC]	D→A]	[D→EF]

If all merge with single Table:

E_1RE_2

A	B	C	D	E	F
a_1	-	-	d_2	-	-
a_2	-	-	d_3	-	-
a_3	-	-	d_4	-	-
a_4	-	-	Null	Null	Null
Null	NULL		d_1	-	-

Candidate key = {A, D}

No Candidate key is present, & No primary key

A relational table in which no attribute having "NOT NULL" values are "NOT" allowed in RDBMS . So,

- minimum 2 Relation Table Required and 1 Foreign key.

E_1 E_2 Entity set R Rel set between E_1E_2
With 1:1 mapping and at least one end having total participation.

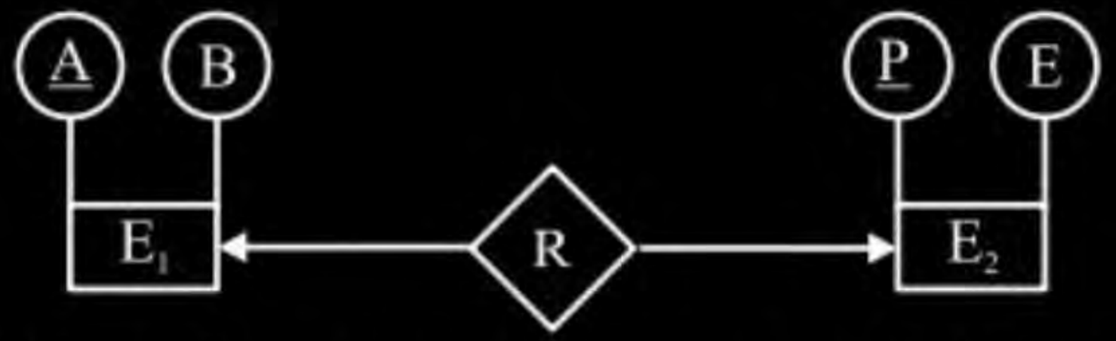
E_1 R E_2 (A B C D E F)

E_1 R E_2 (A B C D E F)

Primary Key

UNIQUE & NOT NULL (Alternative Key)

Q.



70% participation at E_1 end & 40% participation at E_2 end. Which is best possible design?

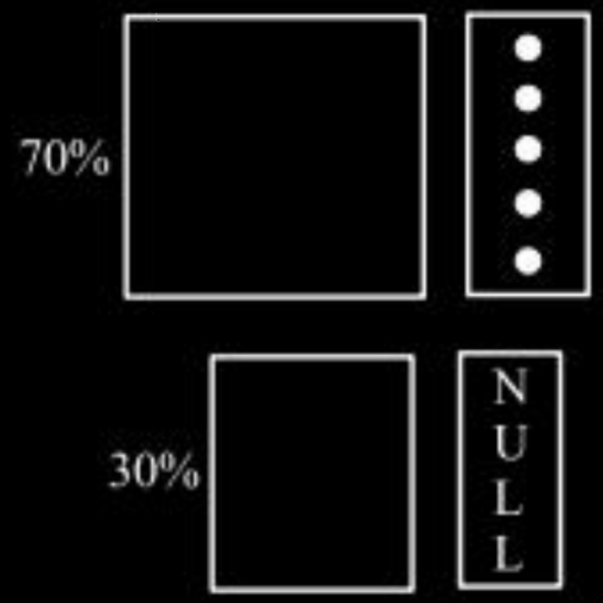
A. E_1 E_2 entity set kept separate with F.K at E_1

B. E_1 E_2 entity set kept separate with F.K at E_2

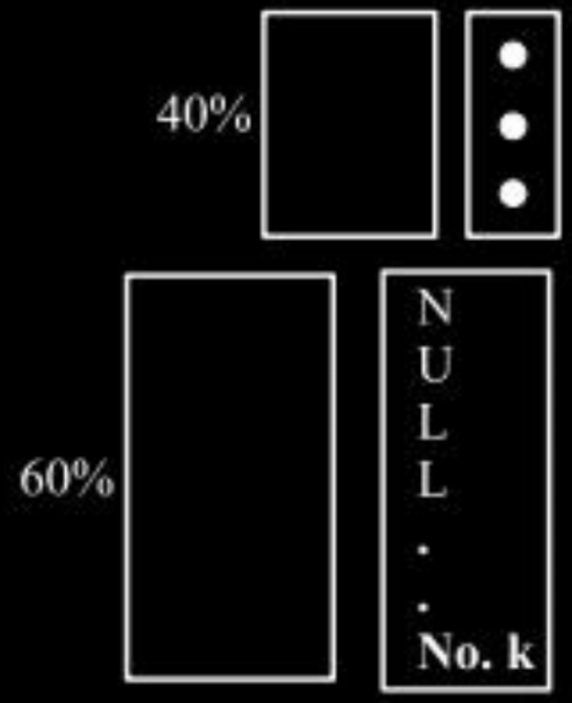
C. E_1 E_2 entity set kept separate with F.K at both E_1 E_2

D. E_1 E_2 merges into single table with No F.K.

(a) E_1 R (A B D) E_2 (D E)

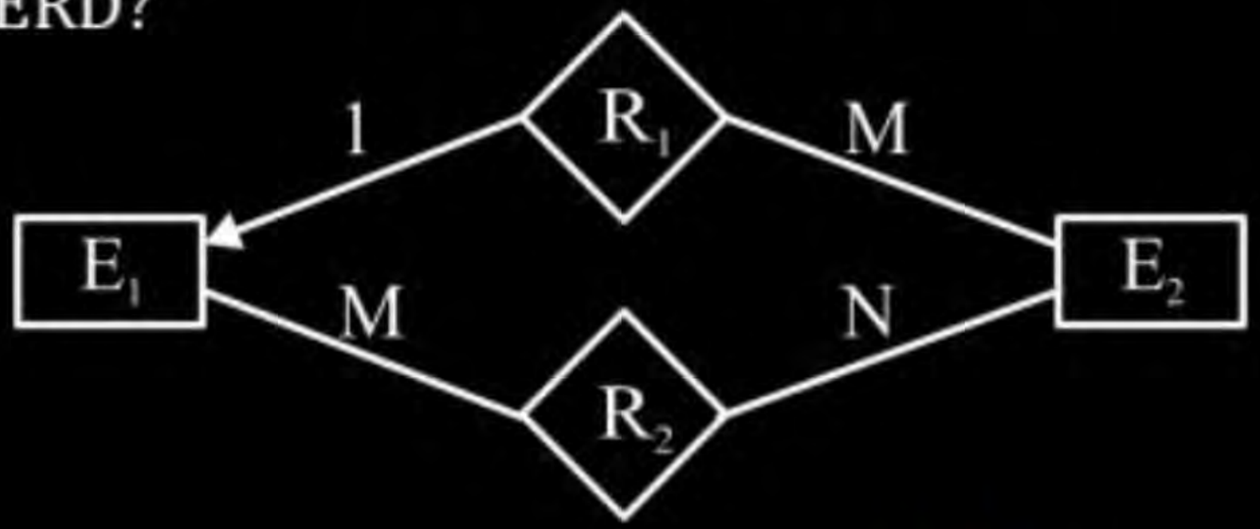


(b) E1(A, B) E_2 (D E A)



Q.

E_1 E_2 entity set R_1 R_2 Relationship set related between E_1 and E_2 with 1:M and M : N mapping min Relational Table required in ERD?



A. 2

B. 3

C. 4

D. 5

Q.

$E_1E_2E_3$ entity set R_1 Relation between E_1 E_2 with 1 : M, R_2 Relations between E_2 E_3 with M : 1, R_3 Relationship between E_1 & E_3 with M : N

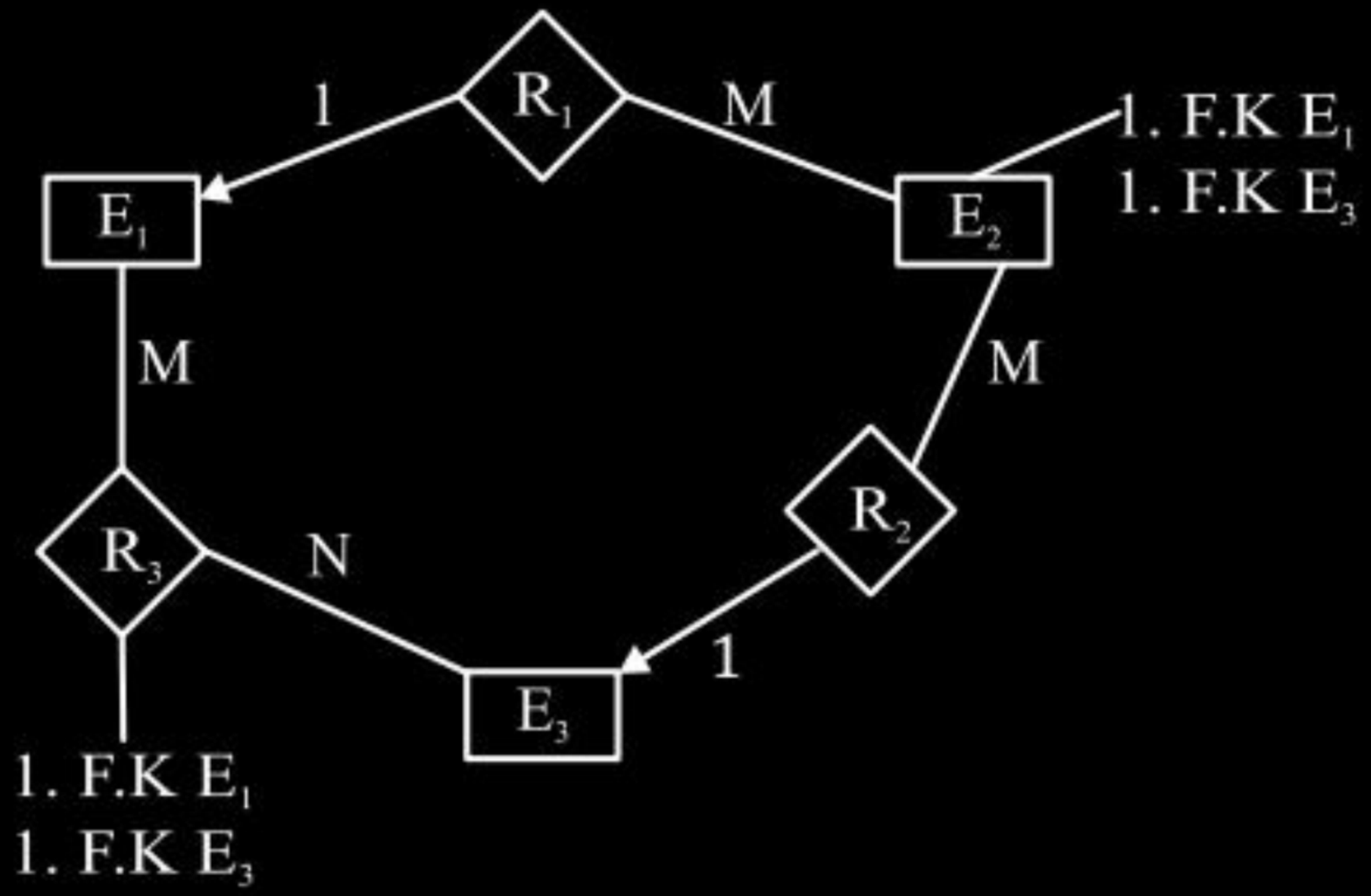
How many minimum relational tables required for given ERD?

A. 2

B. 3

C. 4

D. 5



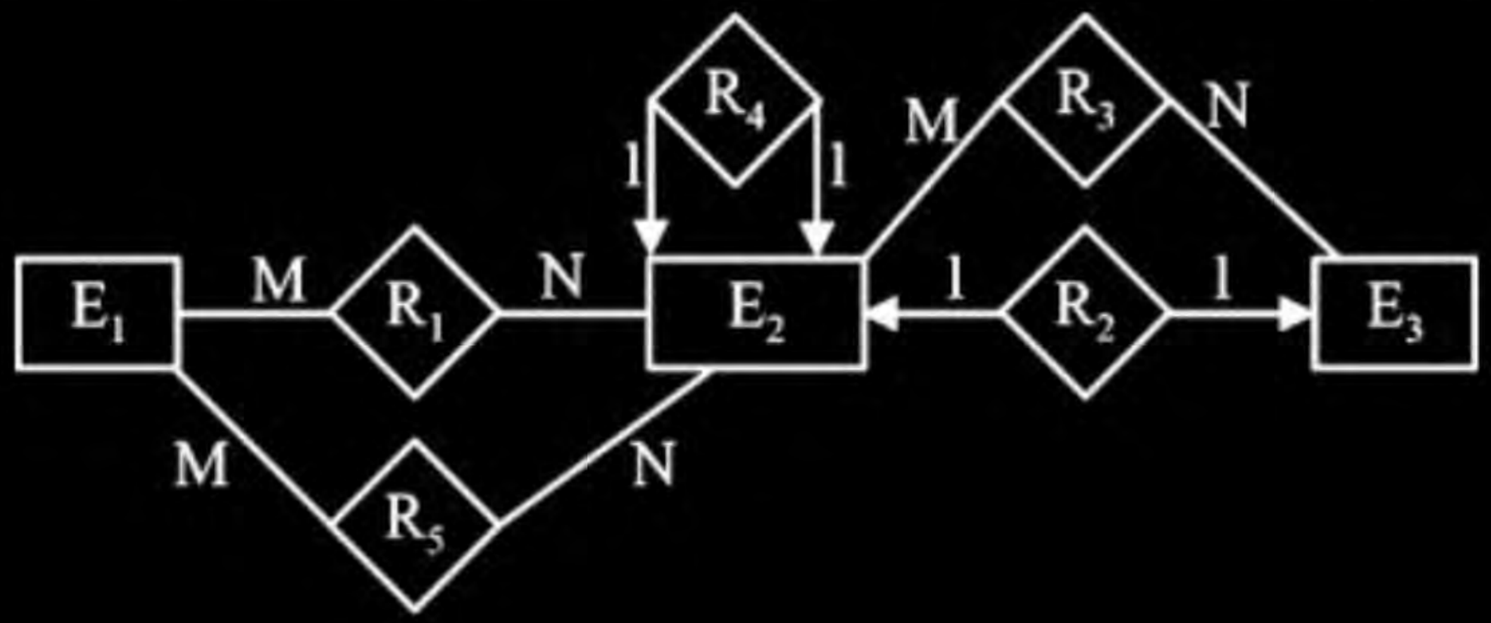
Min 4 Rel Table and 4 Foreign key.

Q.

[NAT]

Consider the following ER diagram

Total number of RDBMS table in the above diagram?



Self Referential Relationship set:

[Recursive entity set]

Entities of entity set (E) related to some other entity of same entity set (E)

} Recursive Entity set

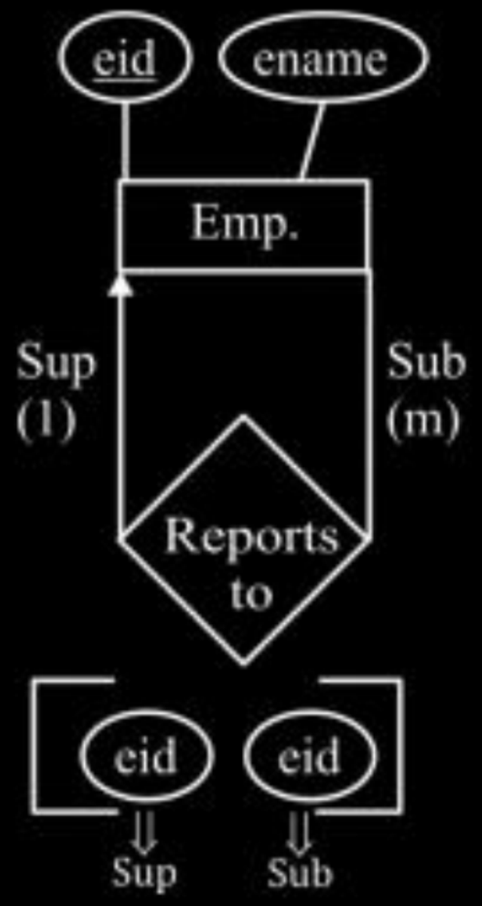
} Self Referential Relationship set.

- Also Binary Relationship set
- 1:M, M :1, M :1, 1:1, N: N be any of the relationship

⇒ Emp entity set.

Reports to Relationship Set Related between Supervisor and Subordinate

(i) Each supervisor can supervise many subordinates and each subordinate reports to one supervisor.



Emp.	(eid		ename)
	e1		–
	e2		–
	e3		–
	e4		–

Reports to	(Sup		Sub)
	e1	–	e2
	e1	–	e3
	e2	–	e4

Emp.	(eid		ename)
	e1		–
	e2		–
	e3		–
	e4		–
Reports to	(Sup		Sub)
	e1	–	e2
	e1	–	e3
	e2	–	e4

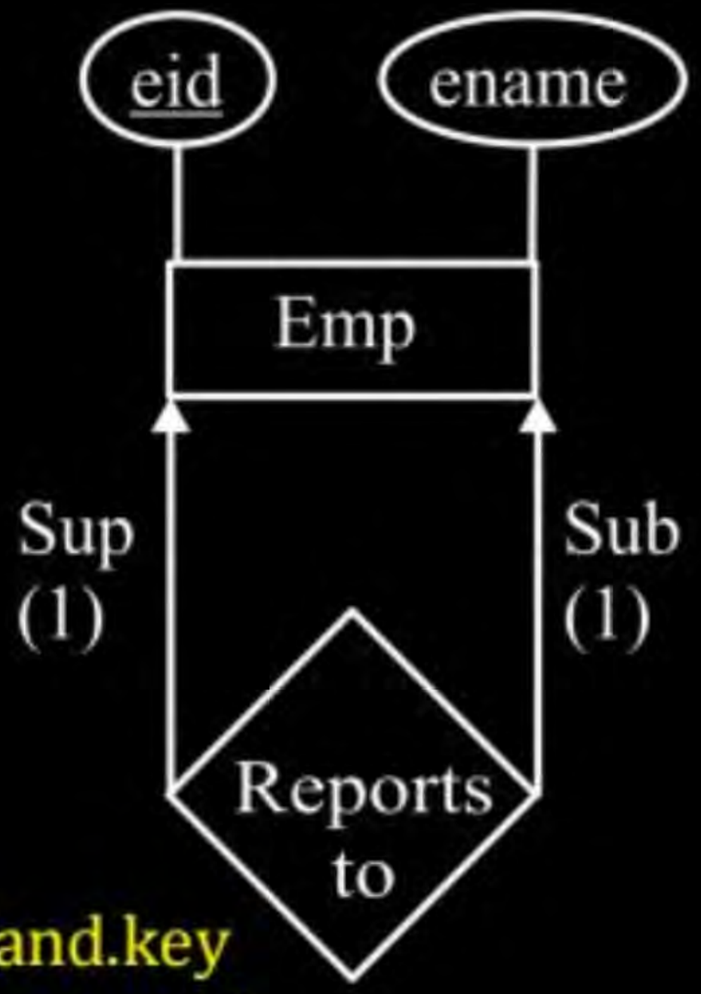
(eid + sub) ⇓ Combine

Cand. Key; {eid}
"Sup" Foreign key Ref.
to same table

Emp. Reports to	(eid	ename	Sup)
	e_1	→	Null
	e_2	→	e_1
	e_3	→	e_1
	e_4	→	e_2

Min 1 RDBMS table required with 1 F.K refer to same table

(ii) Each supervisor can supervise one subordinates and each subordinate reports to one supervisor



Cand.key
{sup, sub}

Emp.

(eid	ename)
e1	–
e2	–
e3	–
e4	–

Reports to

(Sub		Sub)
e1	–	e2
e2	–	e3
e3	–	e4

Emp_Report to **Merge** Emp_Reports to (eid ename Sup)

eid		Sup
e_1	$\rightarrow$	Null
e_2	$\rightarrow$	e_1
e_3	$\rightarrow$	e_2
e_4	$\rightarrow$	e_3

Can. Key : {eid, sub}

eid $\Downarrow$ P.K. (eid + sub)

sub $\Downarrow$ A.K

"Sup" foreign key Refers to same Table eid.

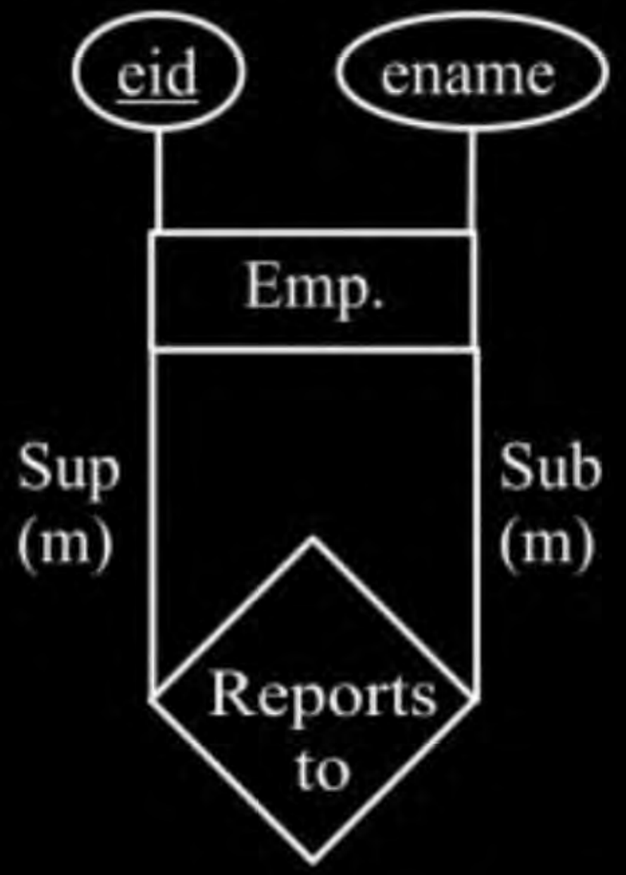
Or one more possibilities

(eid ename sub)

eid $\downarrow$ P.K (eid+sup)

sub $\downarrow$ A.K

(iii) Each supervisor can supervise many subordinate and each subordinate reports to many supervisor.



Emp.

(eid	ename)
e1	–
e2	–
e3	–
e4	–

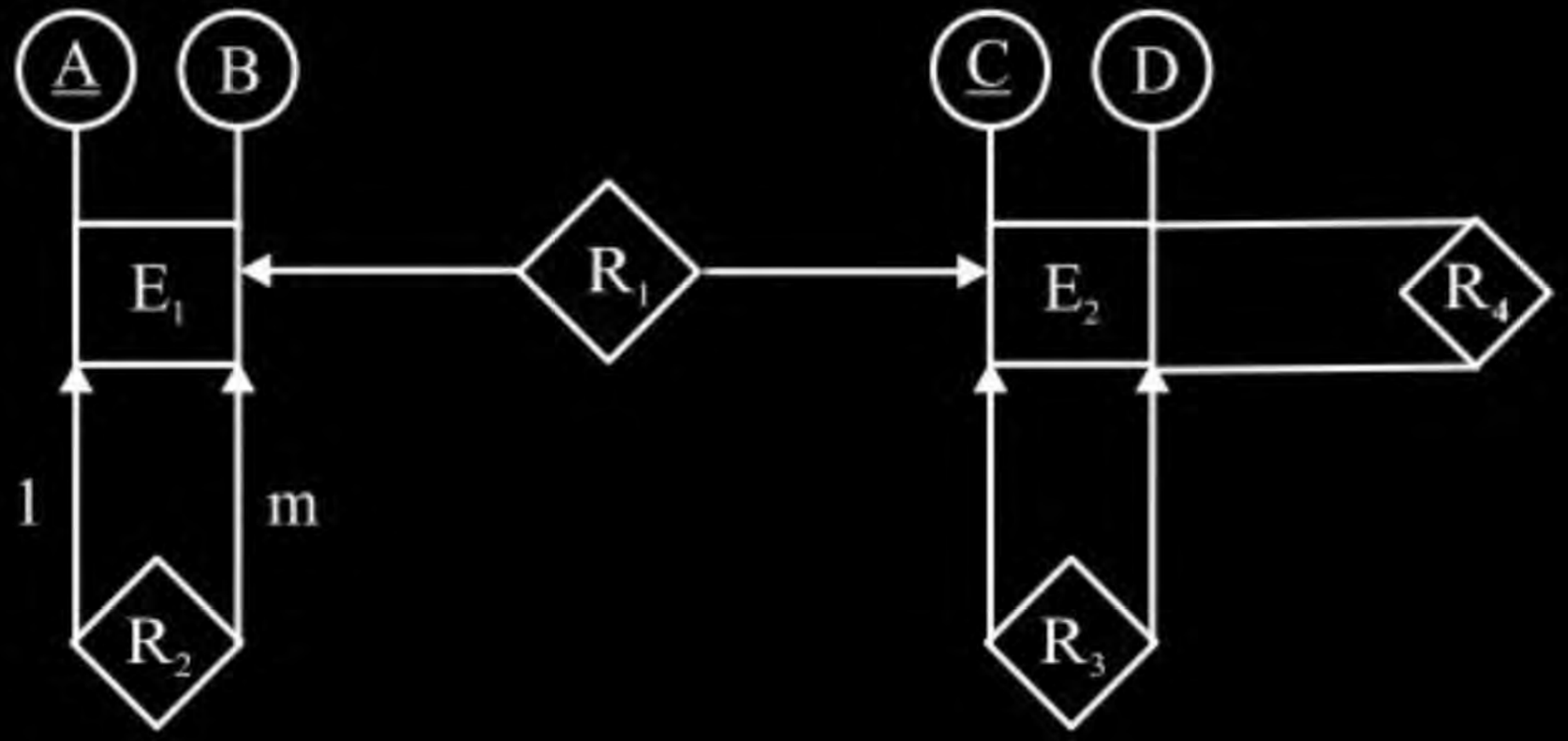
Reports to

(Sub		Sub)
e2	–	e2
e1	–	e3
e2	–	e3

Cand.key
{sup sub}

Minimum 2 relational table and 2 foreign keys.

Q.



(I) $E_1 R_2$ (A B A)

(II) $E_2 R_1 R_3$ (C D A C)

(III) $R_4 C_3 C_4$

Q.

[MCQ]

Consider the following ER diagram:

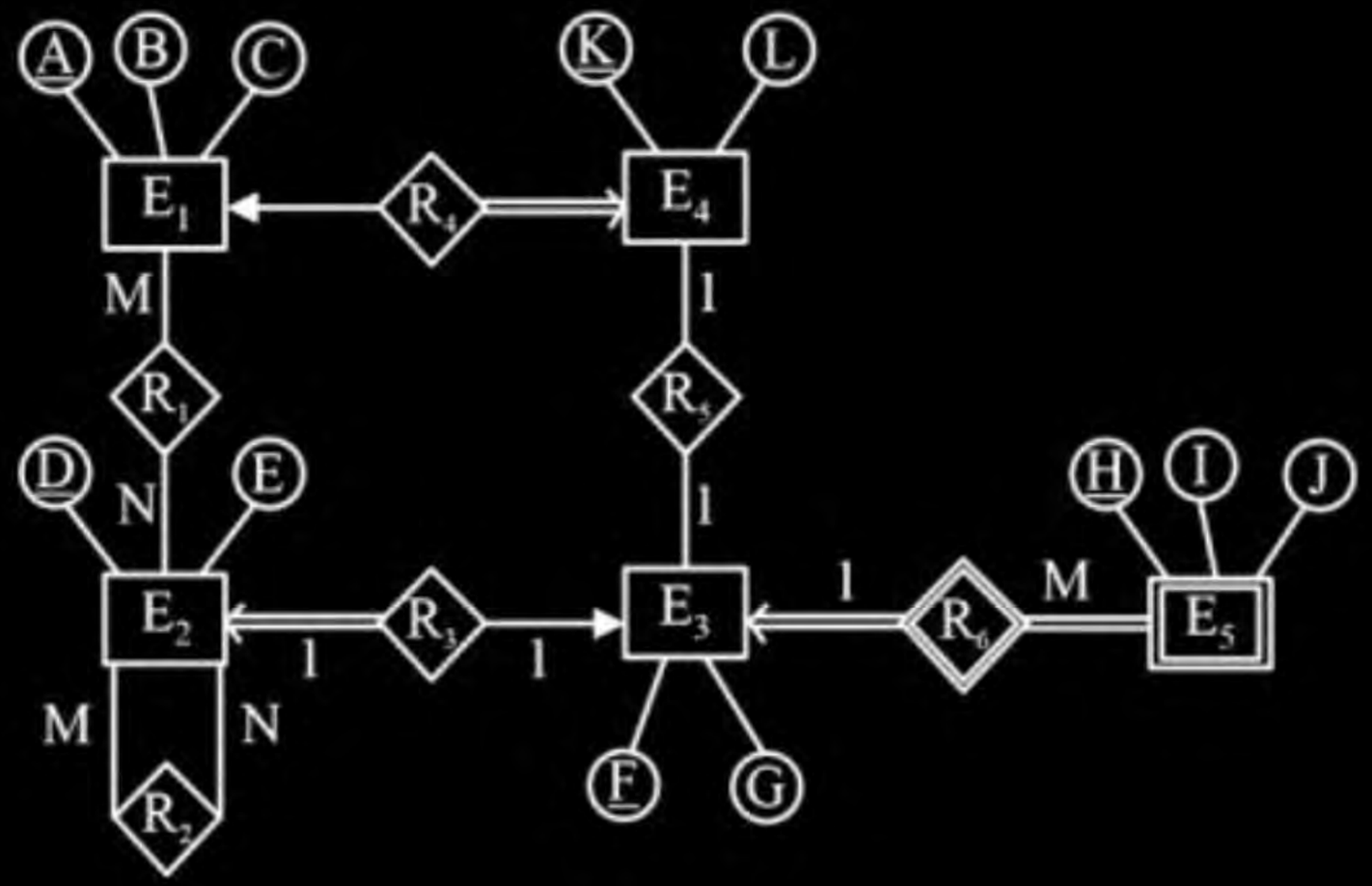
How many total attributes required for the minimized relations of the above ER diagram?

A. 14

B. 15

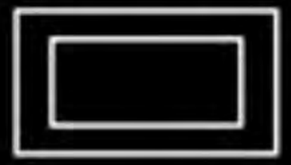
C. 18

D. None of these

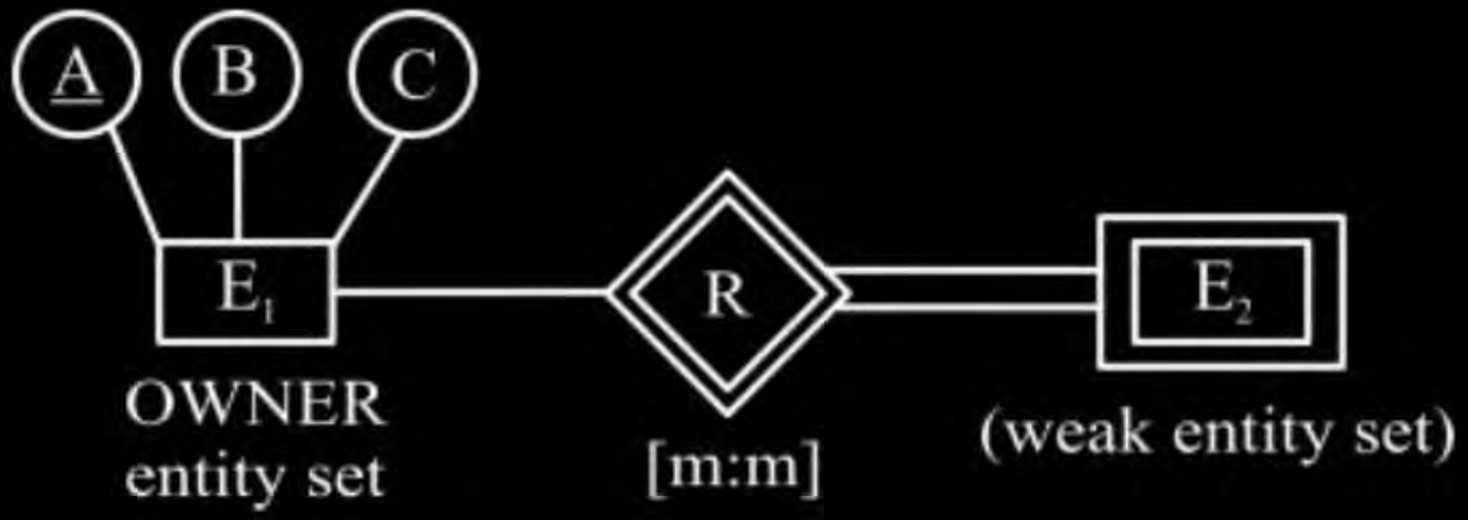


WEAK ENTITY SET:

⇒ Entity set with no key.
[Attributes of weak entity set not sufficient to differentiate entities uniquely]

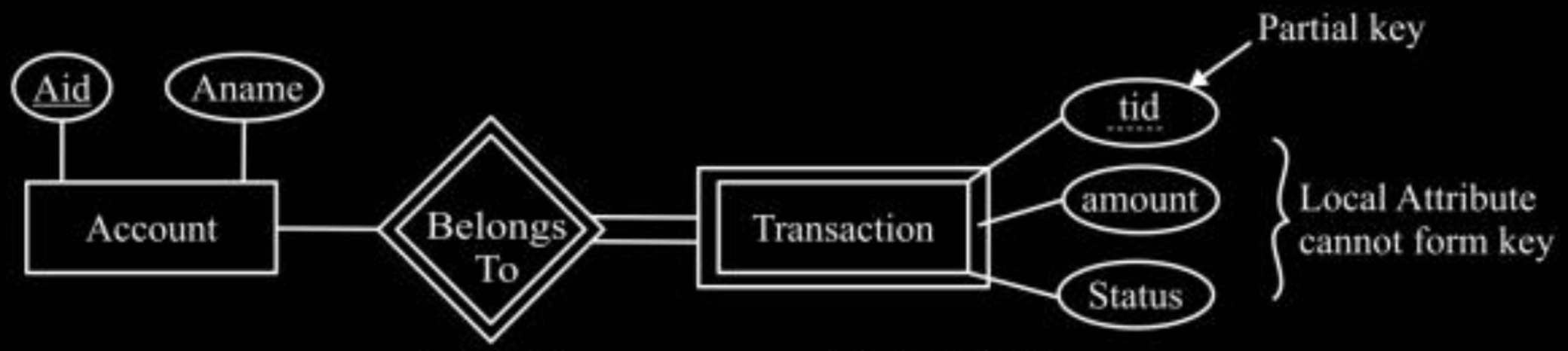
⇒ 

⇒ For each weak entity set there must be OWNER entity set which is strong entity.



⇒ Entities of weak entity set are depending entities.

Ex.



Aid	Aname
A_1	
A_2	
A_3	
A_4	

Aid tid

Aid	tid
A_1	t_1
A_2	t_1
A_2	t_2
A_3	t_2

⇑

[1: m]
Ambiguity

Transaction

tid	amount	Status
t_1	5000	Debit
t_1	4000	Credit
t_2	5000	Debit
t_2	5000	Debit

- Relationship set between weak entity set and identifier entity is also weak relationship set.

- Participation towards weak entity set end must be TOTAL PARTICIPATION.
- Mapping between identifier set must be one: many (1: m)
- RDBMS Design:
  Transaction belongs (<u>Aid tid</u> amount state)
  Account (<u>Aid</u> Aname)

NOTE: Weak entity set and multivalued attributes allowed to represent in ERD, but NOT allowed in RDBMS Table.

Q. In which one of the following Lock Scheme Deadlock cannot occur? [MCQ]

A. Basic 2PL

B. Strict 2PL

C. Conservative 2PL

D. Rigorous 2PL

Q.

Consider the following statement about lock–based protocol

(A) 2 PL (2phase locking) protocol Ensure view serializability

(B) 2PL ensure recoverability &No cascading rollback.

(C) Strict 2 PL ensure recoverability & no cascading rollback.

(D) Strict 2 PL avoids deadlock (not suffering from deadlock).

How many numbers of above statement are correct?

[MCQ]

A 1

B 2

C 3

D 4

Q.

Consider the following Schedule: **[MSQ]**

$r_1(x)\ r_2(y)\ r_2(x)\ w_1(z)\ r_1(y)\ w_3(y)\ r_3(z)\ w_2(y)\ w_3(x)$

which of the following time stamp ordering Not allows to execute schedule using Thomas Write rule time stamp Ordering Protocol?

A. $(T_1, T_2, T_3) = (20, 30, 10)$

B. $(T_1, T_2, T_3) = (10, 20, 30)$

C. $(T_1, T_2, T_3) = (10, 30, 20)$

D. $(T_1, T_2, T_3) = (30, 20, 10)$

Q.

Consider the following statements:

S_1: All strict recoverable schedule are serial.

S_2: All recoverable schedules are conflict serializable.

S_3: All strict schedules are conflict serializable.

S_4: All conflict serializable schedules are free from cascading rollbacks.

Which of the following is true?

(a) Only S_1 and S_4

(b) Only S_2, S_3 and S_4

(c) Only S_2 and S_4

(d) None of these

Q.

Consider the following transaction:

T_1: $R_1(x)\ W_1(x)\ R_1(y)\ W_1(y)$

T_2: $W_2(y)\ W_2(x)$

The number of non-serial schedules between T_1 and T_2 which are serializable?

(a) 2

(b) 13

(c) 15

(d) None of these

Any Doubt ?

THANK YOU!