Database Management

GATE CSE NOTES

Standard : Division : Roll :

Subject : DBMS (Database management system)

INDEX

ER-Model

• Introduction to DBMS :

Types / examples: Text, number, Images, videos, Speech. — **Data** — Data is – (def) (any fact that can be recorded)

↓

1. Text and number (Tradisnal data base) — **Data Base** — Data base is – (Collection of related data)
2. Multimidia data base.
3. Geographi Information system.
4. Realtime data base.
5. Data ware house. (Data huge)

↓

DBMS — DBMS is – Set of programe use to – (Define, construct, manipulate)

Book, pen

(DB + DBMS) ⟶ DBS (Data base system)

(collection of data + set of program)

• Models in DBMS : (Data Base design)

ER model ← **HL or conceptual model** (high level) - CL

↓

Relation model ← **Representational or implementation** - RL

drawing chart = ←

↓

actual work ← **LL or physical data models** (low level) - PL

- **Introduction to ER model** = (Entity-relationship model)

In ER-model - 3 thing should know

Person — Thing — Entity ✓

name, age — properties — Attributes ✓

works for — assocation — Relationship ✓

→ Entity type — Entity

PERSON (Age, Nam, add) — (26, Rakesh, ...)

ER ✓ → Relation table → Hard disk

- **Attributes** =

composite vs simple attribute.
single valued vs multivalued attribute.
stored vs derived attribute.
complex attribute. → (Composite + Multivalued).

Entity → Person.
Attribute → Name, Age, Add, P.NO; Date of Birth

Name: FN, MN, Last Name — composite

Age — single valued
Add — multivalued
P.NO — multivalued
Date of Birth → Age — derived

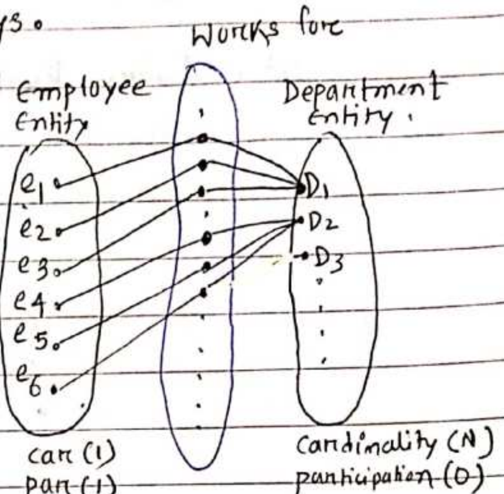
Add → (- - - -) ⋮ (- - - -) } complex attribute.

• **Relationships** =

→ association b/w entitys.

[example-1] (1 to many)

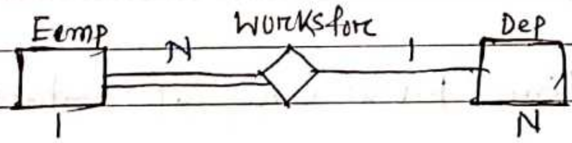
Requirement Analysis: Every employe works for a Department, and a dep can have many employe.
(RA)
New dep need not have any employe.



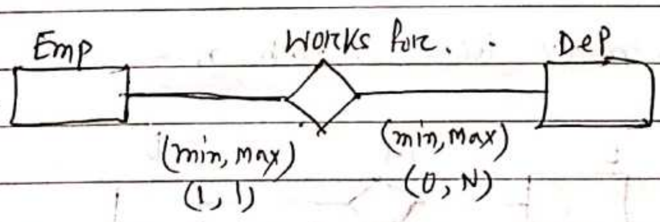
Degree: '2'

Structual Constrain:
- cardinality ratio: Max relationship entity can participate.
- participation or exsitence: → min relationship entity can paricipate.

• Entity represented by - ▭
• Relationship " - ◇

① 

(cardinality ratio / single line / Doubleline representation of ER Diagram.)

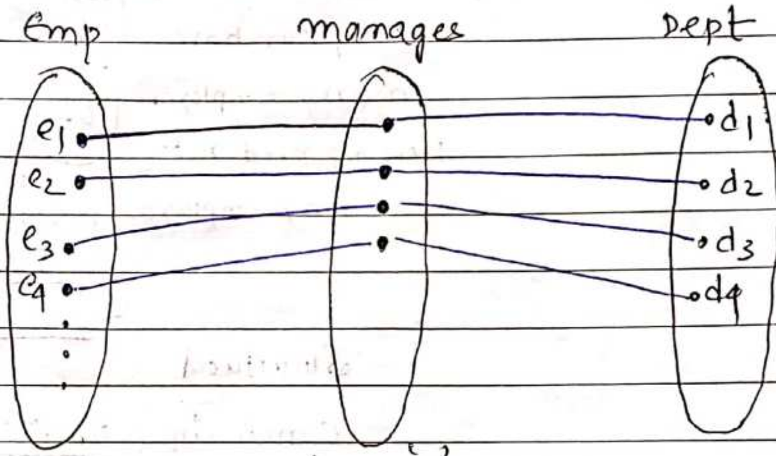
② 

(Min Max represntation of ER Diagram)

most info represent by ↗

example - 2 (1 to 1)

RA : Every Dept should have a manager and only one employee manages a dept and an employee can manage only one dept.

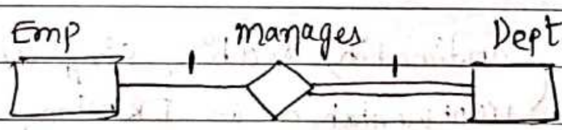


degree = '2'

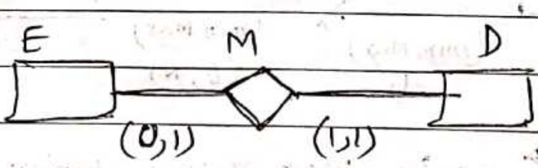
cardinality : 1 1

participation : 0 (P) 1 (Total)

① single line / Double line Representation –

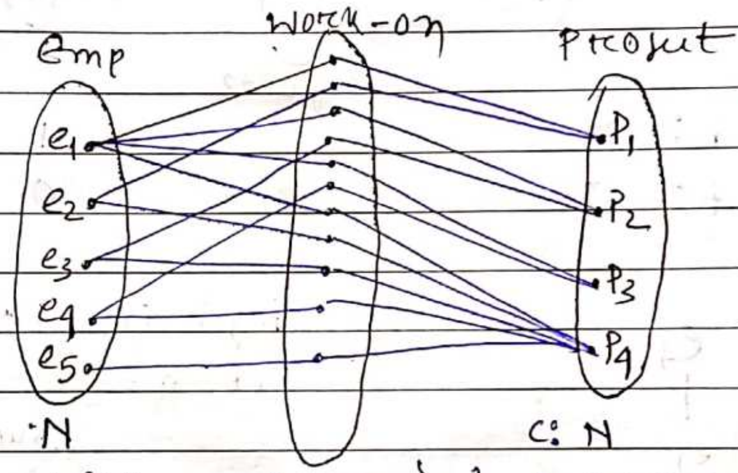


② min max representation –



example-3 (many to many)

RA: Every employ is suppose to work atleast on one project and he can work many project. Similarly a project ~~have~~ can have many employees and every project is suppose to have one employee.



C: N   C: N

P: 1 (T)   degree = '2'   P: 1 (T)

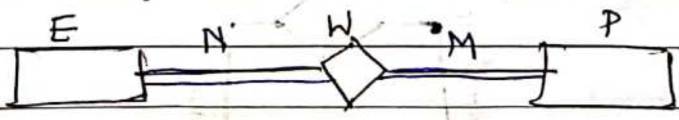
$P_1 = e_1\ e_2$

$P_2 = e_1\ e_3$

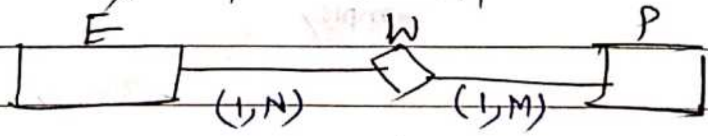
$P_3 = e_1\ e_4$

$P_4 = e_1\ e_2\ e_3\ e_4\ e_5$

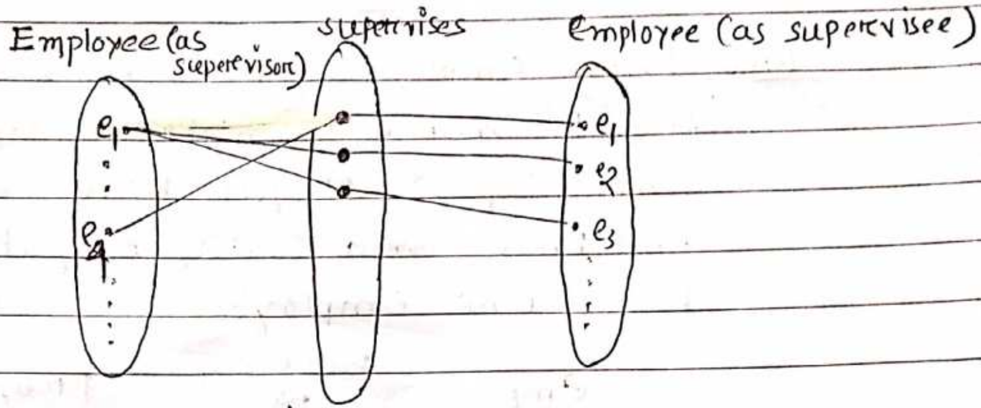
① single line / Double line representation —



② Min Max representation —



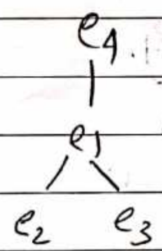
example-4 (Recursive relationship)



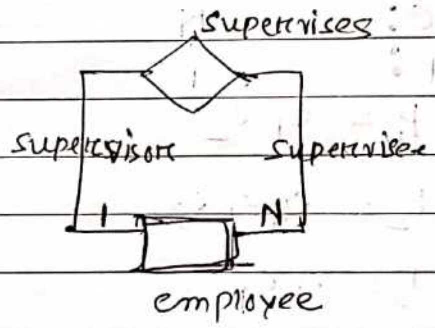
degree=2

card : N 1

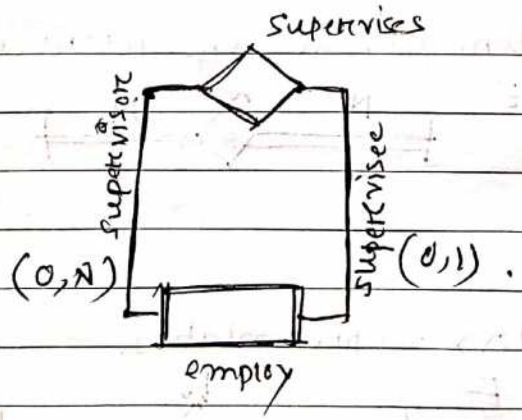
part : 0 0



(1) Single line/Double line representation -

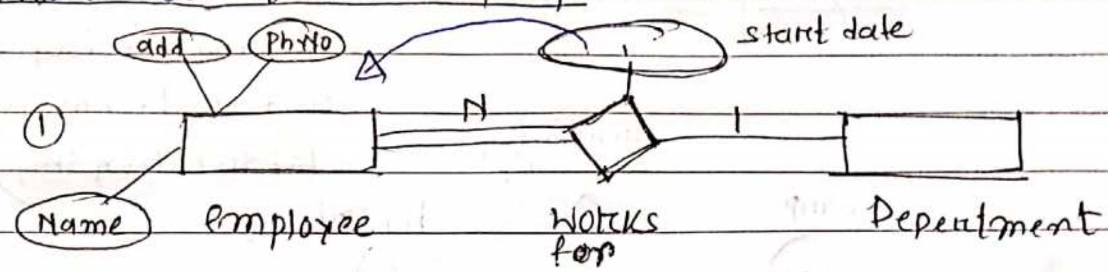


(2) Min Max representation of ER Diagram -

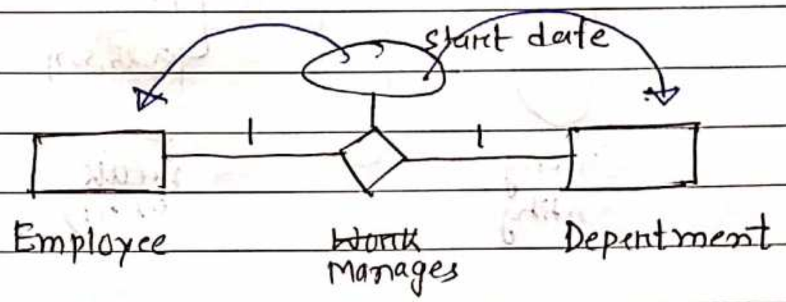


- Attributes to relationships —

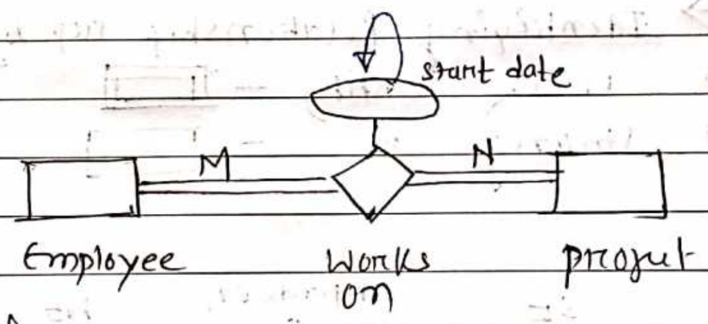
(1)



(2)



(3)



→ In this case Attribute can't shift any side, it will associate with relationship.

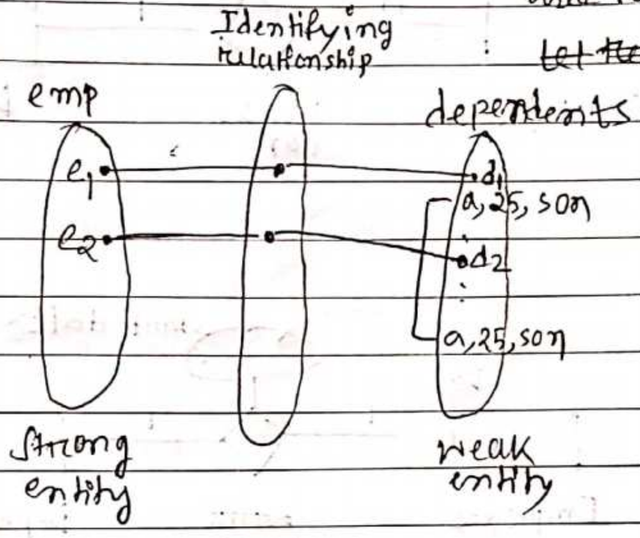
- Weak entity —

→ Whenever any entity not having key attribut then such entity called weak entity.

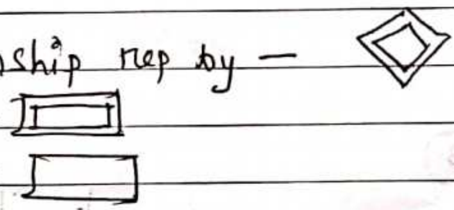
→ ~~wht~~ entity having key attribut, then such entity called strong entity. (like, Id no, Phone.NO, Reg NO).

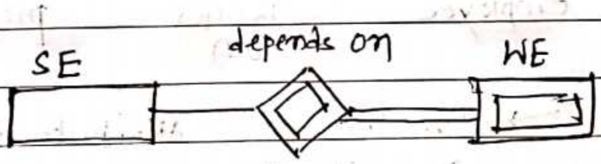
Identifying Relationship :

(~~spatial~~ Special Relationship used to Identifying weak ~~let relationship~~ entity)



→ Identifying Relationship rep by —
→ Weak entity —
→ strong " —





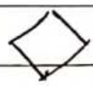
→ Weak entity participation ~~on~~ in Identifying relationship is total.

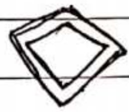
→ And Every Total Participation does't say that the entity is weak.

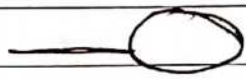
ER-diagram notations –

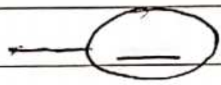
1. Entity – 

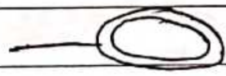
2. Weak Entity – 

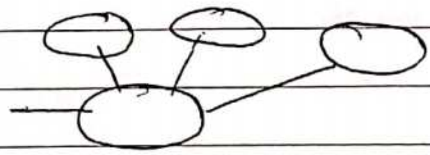
3. Relationship – 

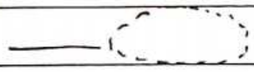
4. Identifying relationship – 

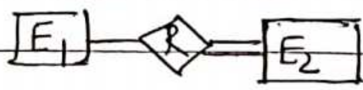
5. attribute – 

6. Key attribute – 

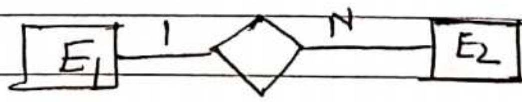
7. Multivalued attribute – 

8. composite attribute – 

9. Derived attribute – 

10. Total participation of E_2 in R – 

11. cardinality ratio $E_1 : E_2 = 1 : N$ –



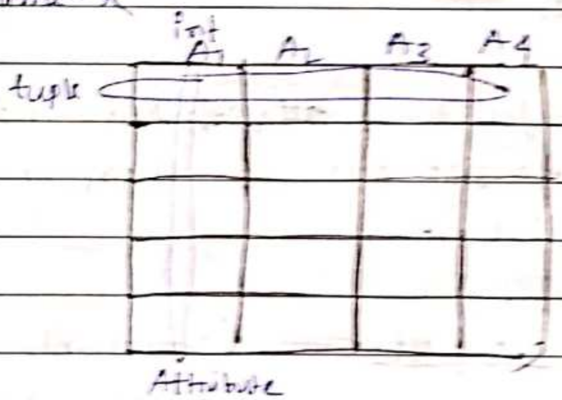
Relational Database Model

- Introduction to Relational model =

- Terminology of Relational database =

relation — (table)
tuple — (row)
attributes — (column)
domain(D) — (Set of value) — (Details)
Relation Schema — $R(A_1, A_2, A_3, A_4)$
degree of relation — (no. of attribute have in table)
relation state —
Intension — Some time relation schema is called Intension.
extension — Table itself.
current relation state → (Elements present in table at any given point of time.)

table name - R



$$R(A_1, A_2, A_3, A_4) \text{ — relation schema}$$

$$\begin{array}{cccc} A_1 & A_2 & A_3 & A_4 \\ \downarrow & \downarrow & \downarrow & \downarrow \\ D_1 & D_2 & D_3 & D_4 \end{array}$$

$$r(R) \subseteq D_1 \times D_2 \times D_3 \times D_4$$

→ In relation model same two tuple can't place.

- Tuple, tuple value and NULL =

Domain

SN /	SNO	Ph.no
a	1	201201
× a	1	201201

→ Two tuble (row) can't same.

FN ~~MN~~ LN Ph

$\left.\begin{array}{l}\text{not applicable}\\ \text{or}\\ \text{unknown}\end{array}\right\}$ NULL

→ When in tuple (row) some attribute (midle name, phone.NO) are not present, then In such case we use? NULL.

- Relational constraints. = (constraints on Relational database Schema)

4 – restriction (constraint):

(1) Domain Constraints.
(2) Key Constraints. (also called uniqueness constraints)
(3) Entity integrity constraints.
(4) Referential integrity Constraints.

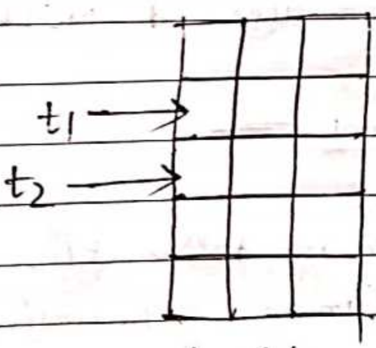
① Domain Constrains — (entire shema of attribute should atomic)

Student

	Name ×			P.NO
S.Nam	FN	MN	LN	
				()()

atomic → Smallest individual unit.
→ every Domain should contain atomic value
→ Comosite attribute not allowed only simple attribute are allowed.
→ multivalued attribute also not allowed.

② Key Constrains — (No two tuple can't be same value)



$t_1 \neq t_2$

→ super key.

(S.No, S Name, marks)
1 Ravi 100
2

Super key ⊆ Attributes.

(2) Key: minimal super-key.

Name	Age	D.No	Street NO

(1) Super Key: Set of Attribut able to findout tuple uniquly

here (Name, Dore no, street no) → Key.
→ In worst case superkey = key.

~~(Relationship b/w Key and superKey)~~

→ any superset of a Key is a superKey.

(S.NO, SName) – SuperKey.
(S.NO, marks) – SuperKey
(S.NO, SName, mark) – SKey.

~~→ In a table m more then one minimap~~

→ (3) **Candidate Keys** : If we have more then one key for a table.

(4) **primary Key** : first choosen candidate key.

→ Key may contain more then one attribute.

→ NULL value are not allow in primary Key.

(3) Entity integrity constraints –

→ ~~here,~~ primary Key never allowed NULL value.

(4) Referential ~~Entity~~ integrity constraints –

→ Referential integrity can be define b/w two table or on the same table (in case of recursive relationship)

Project

P.NO		
1		
2		
3		
4		

Employ

Foreign Key F.K

ENO	Ena	Dip.NO	PR.NO
1	a	1	4

Department

Dip.NO	
1	
2	
3	
4	

→ A table can contain more than one FK (Foreign Key).

- Actions upon constraint violations –

Operations of Data Base –
(I) insert. — (Reject)
(II) Delete.
(III) update. (modify)

Constraints –
(I) Domain
(II) Key
(III) Entity Integrity
(IV) Referential Integrity

(I) insertion :
→ When any of the constraints fail while insert a key are tuple then ~~to~~ Reject the Insertion completly. (all constraints violated by ~~deletion~~ insertion)

(II) Deletion :
→ Domain constraints not violated by Deletion.
→ Key constraints also not " " ".
→ Entity Integrity " " " " "
→ (Only Referential Integrity violated by Deletion.)

Depertment

	D.NO	
x	1	
	2	

Employee

E.na	E.no	D.NO
a	3	1 (N)
		2

3 – solutions – to avoied Refere. Integrity violations –
→ Reject action. (Deletion not possible)
→ cascade (Delete a from both ~~cased~~ table)
→ set NULL or some other value.

(III) Update :
→ combination of Delete and Insert.
(Delete a old ~~value~~ tuple & ~~put~~ Insert new tuple)

- ~~Example - 1~~

~~Counting the number of Super Keys possible~~

- Counting the number of super keys possible:

Example - 1

given relation,

$R(A_1, A_2, A_3, \dots A_n)$

candidate key $= \{A_1\}$
(minimal key)

$\rightarrow$ $R(A_1, A_2, A_3)$
$CK = A_1$ 2×2

$SK = \{A_1, \underline{A_1A_2}, \underline{A_1A_3}, \underline{A_1A_2A_3}\}$
$= 2^{3-1} \quad (2^{n-1})$

$R(\cancel{A_1}, A_2, A_3 \dots A_n)$
$2 \times 2 \times 2 \times \dots \times 2$

no. of $SK = 2^{n-1}$

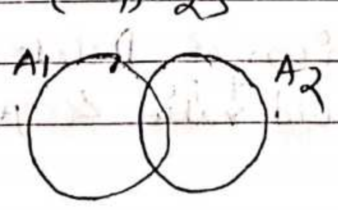
Example - 2

give, $R(A_1, A_2, A_3, \dots A_n)$
$CK = A_1A_2$

$R(\underline{A_1, A_2}, A_3, A_4, \dots A_n)$
$2 \times 2 \times \dots 2$

$\rightarrow$ $CK = A_1A_2$ — $SK = 2^{n-2}$
$CK = A_1A_2A_3$ — $SK = 2^{n-3}$

$CK = \{A_1, A_2\}$ — $[SK(A_1) + SK(A_2) - SK(A_1A_2)]$



$= 2^{n-1} + 2^{n-1} - 2^{n-2}$

Example - 3

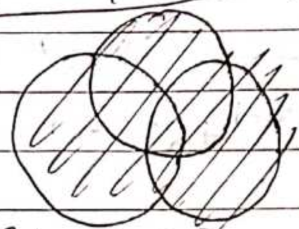
given, $R(A_1, A_2, A_3, \dots A_n)$

(i) $CK = \{A_1, A_2, A_3\}$

$\rightarrow SK(A_1) + SK(A_2) \cancel{+ SK(A_3)} - SK(A_1A_2A_3)$

$\rightarrow 2^{n-1} + 2^{n-2} \cancel{+ 2^{n-2}} + 2^{n-3}$

$SK \rightarrow 2^{n-1} + 2^{n-2} + 2^{n-3}$



(ii) ~~$CK = \{A_1, A_1A_2\}$~~

(iii) $\rightarrow CK = \{A_1A_2, A_3A_4\}$

$\rightarrow SK(A_1A_2) + SK(A_3A_4) - SK(A_1A_2A_3A_4)$

$\rightarrow 2^{n-2} + 2^{n-2} - 2^{n-4}$

$SK \Rightarrow 2^{n-1} - 2^{n-4}$

(iv) $CK = \{A_1A_2, A_1A_3\}$

$SK = SK(A_1A_2) + SK(A_1A_3) - S(A_1A_2A_3)$

$= 2^{n-2} + 2^{n-2} - 2^{n-3}$

$SK = 2^{n-1} - 2^{n-3}$

$\boxed{\text{Example - 4}}$

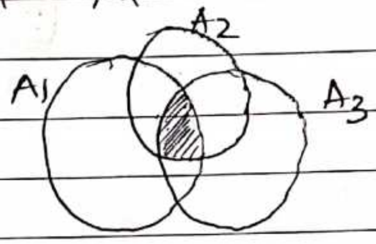
$$R(A_1, A_2, A_3, \ldots A_n)$$

$$CK = (A_1, A_2, A_3)$$

$\rightarrow SK = SK(A_1) + SK(A_2) + SK(A_3) - SK(A_1A_2) - SK(A_2A_3) - SK(A_1A_3) + A_1A_2A_3$

$SK = 2^{n-1} + 2^{n-1} + 2^{n-1} - 2^{n-2} - 2^{n-2} - 2^{n-2} + 2^{n-3}$

$= 2 \cdot 2^{n-1} - 2 \cdot 2^{n-2} + 2^{n-3}$



$= 2^n - 2^{n-1} + 2^{n-3}$

No of super key possible.

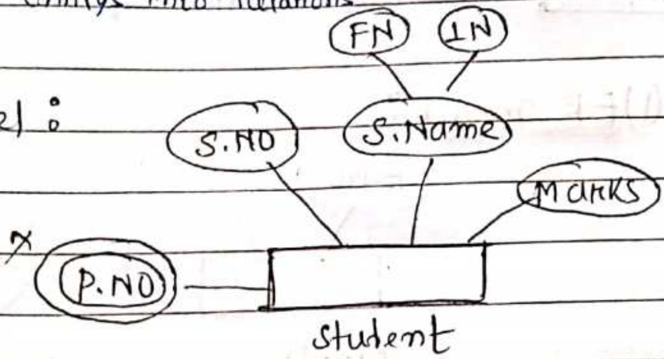
$R(A, B, C, D)$
$CK = (A, BC)$

$SK = S(A) + S(BC) - S(ABC)$
$= 2^3 + 2^2 - 2^{4-3}$
$= 8 + 4 - 2$
$= 10$ super key possible.

Conversion of ER model to Relation model

Step : 1 :- "convert entitys into relations"

ER model :



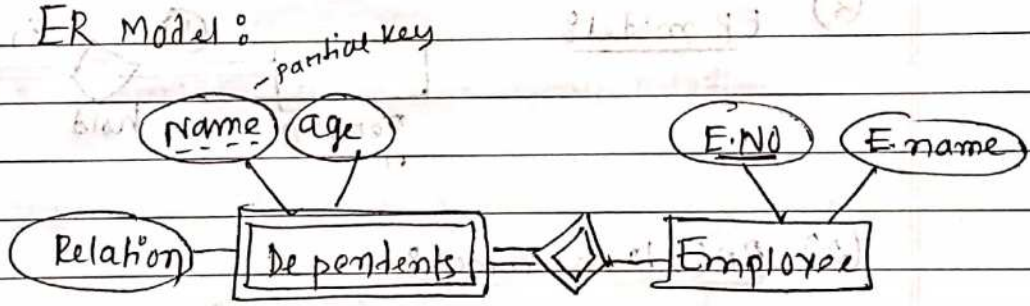
Relation model :

student

S.NO	Marks	FN	LN

Step-2 :- "convert week entity into Relation"

ER Model :



Relational model :

Dependent

FK

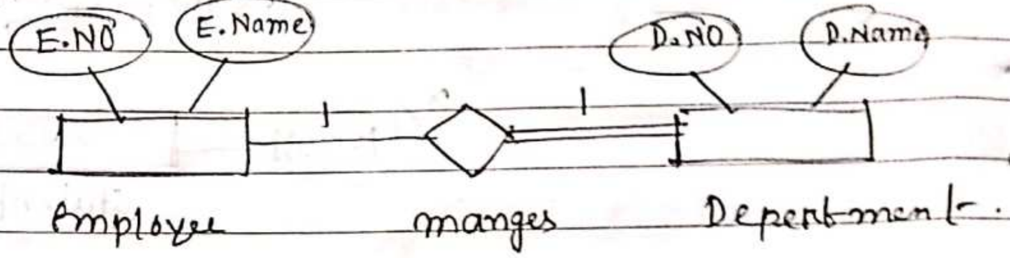
E.NO	Name	Age	Relation

Employee

E.NO	E.Name

Step-3 : "Convert Relationship into Relation"

(1) ER model :



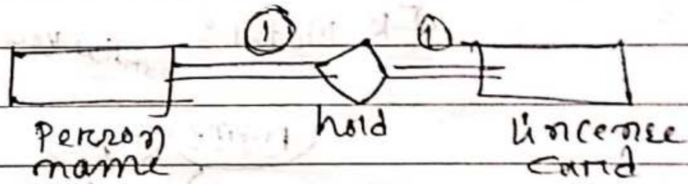
(1) Relation model :

FK

E.NO	E.Name

D.NO.	D.Name	Manges

(2) ER model :



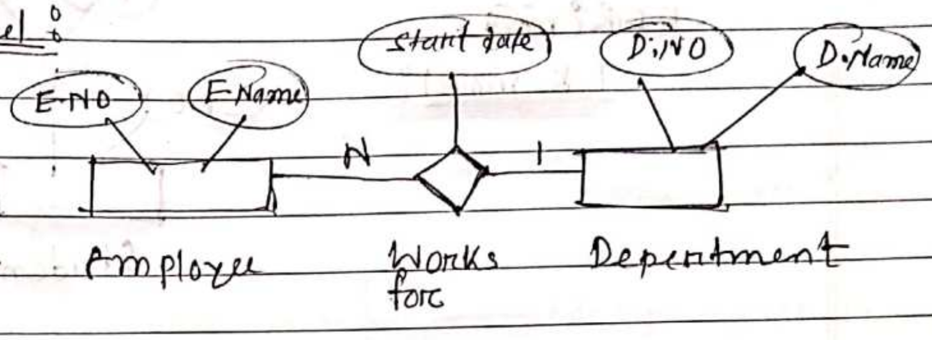
(2) Relation model :

(both side participate totaly)
Combine them in one Relation (table)

person name	lincence

Step - 4 : "convert one to n relationship in table (Relation)"

ER Model :



Relation model :

Employee

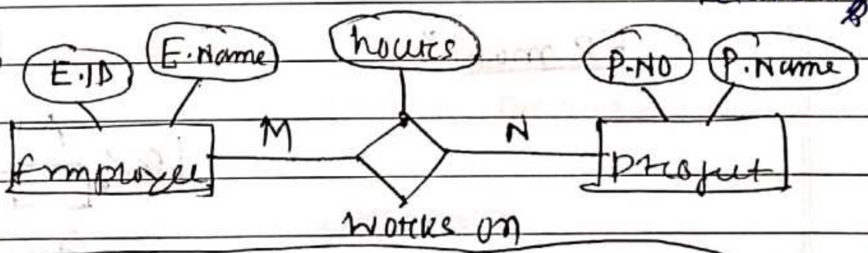
E.NO	E.Name	D.No	SD

Department

D.NO	D.Name

Step -5 : "convert many to many relationship into Relation"

ER model :



E.ID	E.Name
1	a
2	
3	
4	

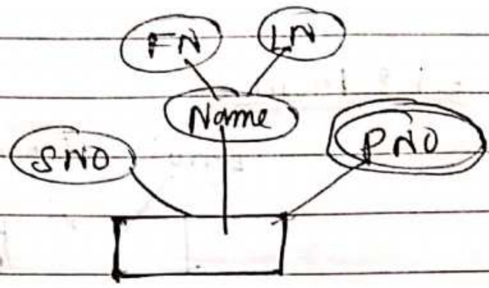
P.NO	P.Name
1	b
2	c
3	d
4	e

E.ID	P.NO
1	1
1	2
1	3
2	1

$\boxed{\text{Step-6}}$: "multivalued attributes"

~~ER model~~

ER model :



Student .

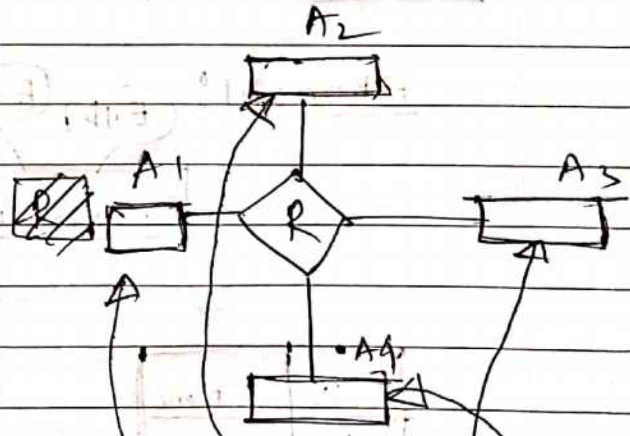
Relation model :

S.NO	FN	LN	~~P.NO~~
1			
2			
3			
4			
5			

S.NO	P.NO
1	P_1
1	P_2
1	P_3
	:

$\boxed{\text{Step-7}}$ "n-ary Relation-Ship"

ER model :



Relationship :

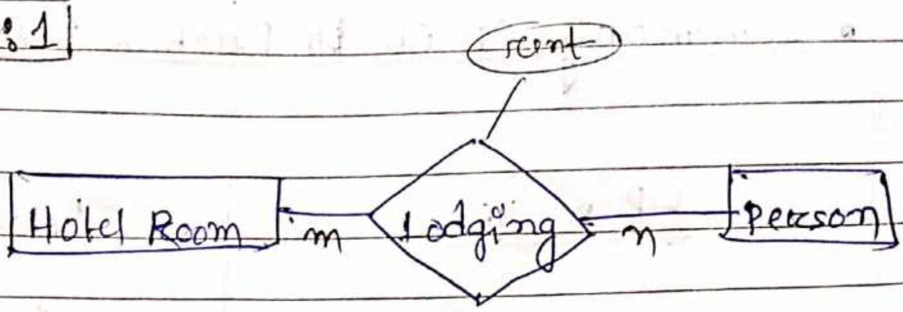
A_1	A_2	A_3	A_4

• Summary Of ER to Relation Data Base Conversion:

ER MODEL	Relational model
Entity type	"entity" relation
1:1 or 1:N relationship type	Foreign key (or relation)
M:N relationship type	relationship "relation" + 2 FKs
n-ary relationship type	Relationship "relation" + n FKs
Simple attribute	Attribute
composite attribute	set of simple Component attribute.
Multivalued attribute	Relation (table) and Foreign key.
value set	Domain
Key attribute	primary Key

g- 2005

Question : 1



loding is a many to many relationship. Rent, payment to be made by person occupying different hotel rooms should be added as an attribute to -

a) hotel ~~b)~~ lodging c) person d) none,

→ If relationship is many to many we can't go any side.

g- 2012

Question : 2

Given the basic ER and relational models, Which of the following is incorrect?

a) An attribute of an entity can have more than one value.

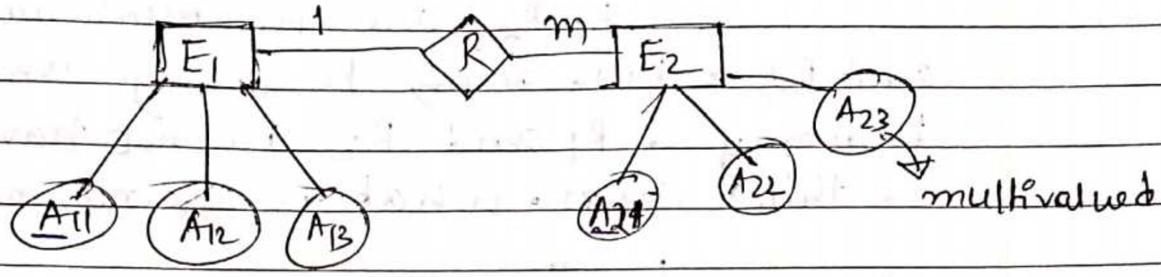
b) An attribute of an entity can be composite.

c) In a row of a relation table, an attribute can have more than one value.

d) In a row of a relational table, an attribute can be have exactly one value or a NULL value.

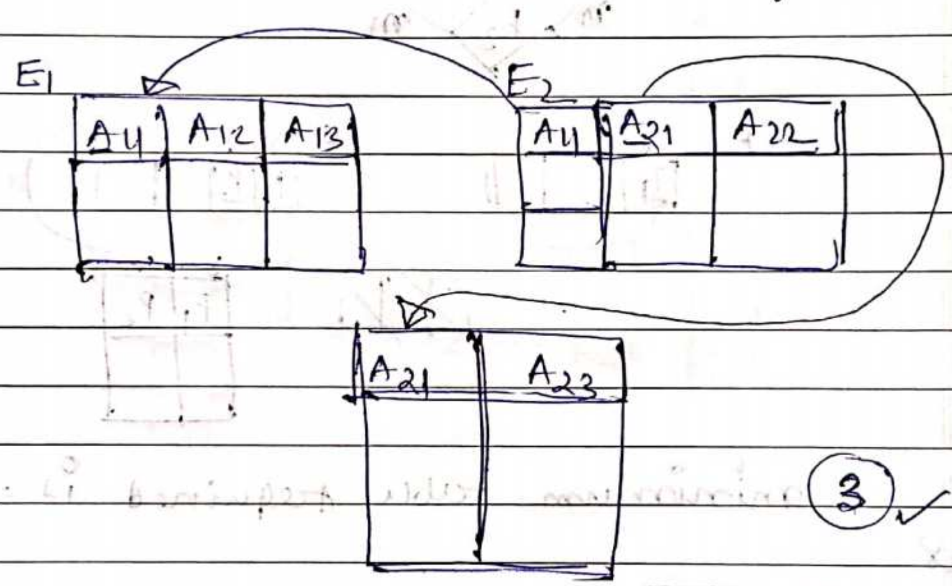
g-2004

Question : 3



minimum number of table = ? (3)

→



(3) ✓

g-2012

Question : 4

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)

A	C
~~2~~	~~4~~
3	4
4	3
5	2
7	2
9	5
6	1

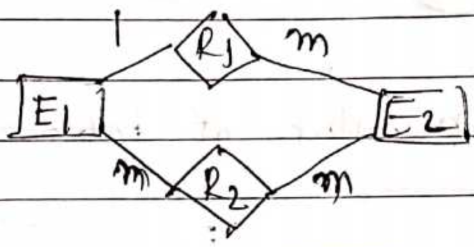
(a) (3,4) and (6,4)
(b) (5,2) and (7,2)
(c) (5,2) (7,2) and (9,5) ✓
(d) (3,4) (4,3) and (6,4)

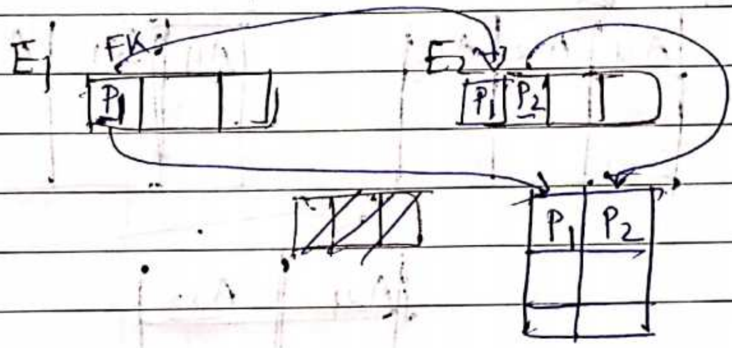
G-2005

[Question : 5]

E_1, E_2 - two entities,
R_1, R_2 - are two relationship b/w E_1 and E_2. R_1 is ~~many~~ one to many and R_2 is many to many. R_1 and R_2 does not have any attributes of their own. What is the number of tables?

→

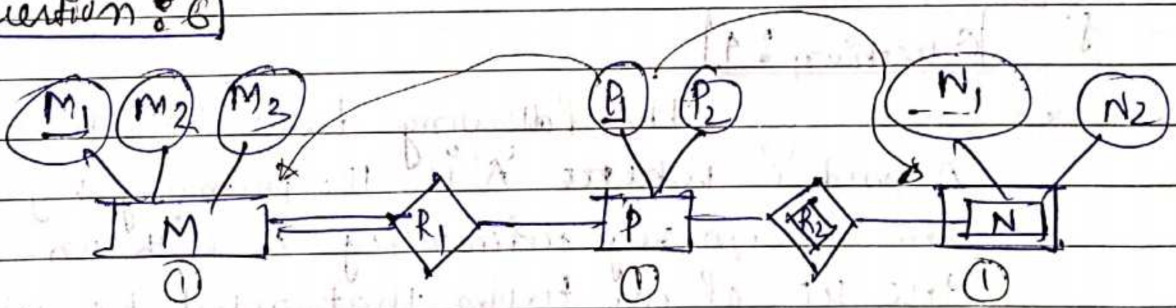




minimum table required is = 3

G-2008

[Question : 6]



The minimum number of tables needed -

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

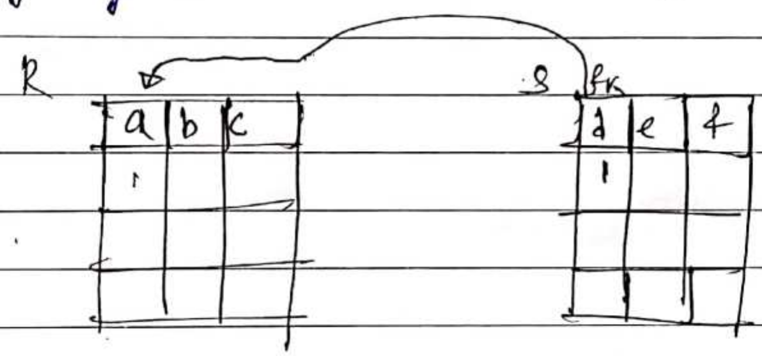
Which of the following is a correct attribute set for one of the tables for the correct answer of the above question -
(A) $\{M_1, M_2, M_3, P_1\}$ (B) $\{M_1, P_1, N_1, N_2\}$ (C) $\{M, P, N\}$ (D) $\{M_1, P_1\}$

Question: 7

let R(a, b, c) and S($\overset{fk}{d}$, e, f) be two relations 'd' is foreign key ~~for~~ of S that refers to the primary key of 'R'. Consider the following four operations on 'R' and 's'.

a) Insert into R. ×c) Delete from R
×b) " " S d) " " S

Which of these can cause violation of referential integrity constraint - (b, c).

→ 

NORMARLISATION

• Introduction of Normalisation =

Emp.

EID	EN	DID
1	a	1

Dep

DID	D Name	SID
1	CS	

EID	EN	DID	DN	SID			
1	a	1	CS	10			
2	b	1	CS	10			
3		1	CS	10			

anomalies (Problems)

— Insertion.
— Deletion.
— Update.

This are all the Problem which will Occure If you try to reduce the over head of dering the Data Base by combining all the table.

Solution =

Divide the table as smaller table as possible. To avoide anomalies the idea size of table is two attribut.

EID	EN

Divide ~~taf.~~ table into small small table which will ~~control less n~~ contain less number of attributes in such a way that dering not contain all this problem, In order to do that particular process which is called as NORMALISATION.

NORMALISATION (used)

→ Functional dependency (FD) } used to evalute formaly.
→ Candidate key (CK).

- Introduction to FD :

A	B	C
1		
2	a	b
3		
4		

$A \to BC$
2 ab

if $t_1(A) = t_2(A)$
then
$t_1(BC) = t_2(BC)$

A	B	
a	1	
a	1	
b	2	
b	2	

$A \to B$
a 1

A	B	C
a	1	d
a	1	e
a	1	f
a	1	g
b	2	h

(reduce redundency) by making a new table.

new table.

A	B
a	1
b	2

reduce redundancy

1 Normal form (NF)
2 NF
3 NF
BCNF

to go 1 NF → BCNF (we need FD & CK)

$\underset{\text{determine}}{X \rightarrow Y}$

- Clarification of FD into 3-Types –

here, X & Y can be ~~set of~~ a collection of attribute or subset of attribute

(1) Trivial	(2) Non Trivial	(3) Semi non-Trivial
$A \rightarrow A$	$X \cap Y = \emptyset$	
$AB \rightarrow A$	$A \rightarrow B$	$AB \rightarrow BC$
$X \supseteq Y$	$A \rightarrow BC$	$X \cap Y \neq \emptyset$
	$AB \rightarrow CD$	

$\boxed{\text{Examples}}$ (FD) =

Rule out the funtional dependencies based on the tables :

①

Eid	Ename
1	a
2	b
3	b

✓ Eid → Ename

✗ Ename → Eid → ruled out.

→ left hand side should unique.

(2)

A	B
1	1
1	2
2	2

$A \rightarrow B$ ✗

$B \rightarrow A$ ✗

(3)

✗ $A \rightarrow B$

✗ $B \rightarrow C$

✗ $B \rightarrow A$

✗ $C \rightarrow B$

✓ $C \rightarrow A$

✓ $A \rightarrow C$

A	B	C
1	1	4
1	2	4
2	1	3
2	2	3
2	4	3

(4)

✓ $A \rightarrow B$

✗ $B \rightarrow C$

A	B	C
1	1	1
1	1	0
2	3	2
2	3	2

(5)

✓ $XZ \rightarrow X$

✓ $XY \rightarrow Z$

✗ $Z \rightarrow Y$

✓ $Y \rightarrow Z$

✗ $XZ \rightarrow Y$

X	Y	Z
1	4	3
1	5	3
4	6	3
3	2	2

(6)

✓ $A \rightarrow B$

✗ $BC \rightarrow A$

✓ $B \rightarrow C$

✓ $AC \rightarrow B$

A	B	C
1	2	3
4	2	3
5	3	3

9-2000

Question-1

Given the following relation instance-

x	y	z
1	4	2
1	5	3
1	6	3
3	2	2

Which of the following functional dependencies are satisfied by the instance-

× (a) $XY \to Z$ and $Z \to Y$
✓ (b) $YZ \to X$ and $Y \to Z$
× (c) $YZ \to X$ and $X \to Z$
× (d) $XZ \to Y$ and $Y \to X$

Question-2

From the following instance of relation schema R(A, B, C) we can conclude that:

A	B	C
1	1	1
1	1	0
2	3	2
2	3	2

× (a) A functionally determines B and B functionally determines C.

× (b) A functionally determines B and B does not functionally determine C.

✓ (c) B does not functionally determine C.

x(a) A ~~det~~ does not functionally determine B and B does not functionally determine C.

• ~~Formal definition of FDS (functional dependencies):~~

• Various usages of FDs:

Operation which have perform on: FDs =

(I) identify additional FDs.
(II) identifying keys.
(III) identifying equivalences of FDs.
(IV) Finding minimal FD set.

TWO method (to perform this operations) =
(1) inference rules.
(2) closure set of attributes.

(1) Inference Rules:

(a) Reflexive: $A \rightarrow B$ if $B \subseteq A$.

(b) Transitive: $A \rightarrow B$ and $B \rightarrow C$ then $A \rightarrow C$.
(c) Decomposition: if $A \rightarrow BC$ then $A \rightarrow B$ and $A \rightarrow C$
(d) Aurgumentation: if $A \rightarrow B$ then $AC \rightarrow BC$.

(e) Union: if $A \rightarrow B$ and $A \rightarrow C$, then $A \rightarrow BC$.

(f) Composition rule: if $A \rightarrow B$ and $C \rightarrow D$, then $AC \rightarrow BD$.

(2) ~~Clouure~~ Closure set of attributes:

FDs: $A \to B$
$B \to D$
$C \to DE$
$CD \to AB$

$A^+ = \{B, D, A\}$
$B^+ = \{B, D\}$
$C^+ = \{D, E, C, A, B\}$
$(CD)^+ = \{C, D, E, A, B\}$
$(AD)^+ = \{B, A, D\}$.

g.-2006

$\boxed{\text{Question - 1}}$

Given,

~~FDS: A → B~~
~~B →~~
FDS: $AB \to CD$
$AF \to D$
$DE \to F$
$C \to G$
$F \to E$
$G \to A$

$(CF)^+ = ?$ $(AF)^+ = ?$
$(BG)^+ = ?$ $(AB)^+ = ?$

→ $(CF)^+ = \{C, F, G, E, A, D\}$

$(BG)^+ = \{B, G, A, C, D\}$

$(AF)^+ = \{A, F, D, E\}$

$(AB)^+ = \{A, B, C, D, G\}$

Determining Candidate Keys =

→ with attribute (A B)
candidate key may possible – (A, B, AB)

→ attribute (ABC)
Candidate key = (A, B, C, AB, AC, BC, ABC)

** → attribute $(A_1, A_2, A_3, \cdots A_n)$
candidate key may possible – $(2^n - 1)$

R(ABCD)

FDS: $A \to B$
$B \to C$
$C \to D$
$D \to A$

① (ABCD)
④ (ABC)(ABD)(BCD)(ACD)
⑥ (AB)(AC)(AD)(BC)(BD)(CD)
④ (A) (B) (C) (D)

$A^+ = \{A\ B\ C\ D\}$
$B^+ = (A\ B\ C\ D)$
$C^+ = (A\ B\ C\ D)$
$D^+ = (A\ B\ C\ D)$

if A is candidate key then (AB) (AC) (AD) (ABC) (ABD) (ACD) (ABCD) will be Super key.

here, fore given ~~Example~~, FDs no. of candidate keyes are '4'.

Question - g - 1996

$R = (A, B, C, D, E, F)$

FDs = $(C \to F,\ E \to A,\ EC \to D,\ A \to B)$

Canditate Key = ?
(A) CD (B) ~~B~~ EC (C) AE (D) AC

$\to$ $(\ \ )^+ = (A\ B\ C\ D\ E\ F)$
↓
CE $= (C, E, F, A, D, B)$ ⟶ Canditate Key.

no. of Super Key $= 2^4$
= 16 superkeys possible which contain CE.

Question - g - 2014

$R = (E, F, G, H, I, J, K, L, M, N)$
and
FDs: {
{EF} → {G}
{F} → {I, J}
{EH} → {K, L}
{K} → {M}
{L} → {N} }

What is Key for "R" ?
(a) {E, F} (b) {E, F, H} (c) {E, F, H, K, L} (d) {E}.

$\to$ $(\ \ )^+ = \{E, F, G, H, I, J, K, L, M, N\}$
↓
EFH

$(EFH)^+ = \{E, F, H, G, I, J, K, L, M, N\}$ ✓

$(EF)^+ = \{E, F, G, I, J\}$ ✗

and,

no of super key possible,

$$2^7 = 2^5 * 2^2$$
$$= 32 * 4$$
$$= 128$$ — SK possible that contain EFH.

$\boxed{\text{Question}}$ - g - 2005

$R = (A, B, C, D, E, H)$

$FD = \{A \to B, BC \to D, E \to C, D \to A\}$

What are the candidate keys on R ?

✗ (a) AE, BE ✗ (b) AE, BE, DE.

✗ (c) AEH, BEH, BCH

✓ (d) (AEH, BEH, DEH)

→

$(\quad)^+ = (A, B, C, D, E, H)$

↓

EH

$(EH)^+ = \{E, H, C\}$ ✗

SK-8

$\boxed{(AEH)^+ = \{A, E, H, B, C, D\}}$ ✓

SK-8

$(BEH)^+ = \{B, E, H, C, D, A\}$ ✓

$(CEH)^+ = \{C, E, H\}$ ✗

SK-8

$(DEH)^+ = \{D, E, H, A, B, C\}$ ✓

$SK = (K_1) + (K_2) + (K_3) - K_1K_2 - K_2K_3 - K_1K_3 + K_1K_2K_3$

$\boxed{\text{Question}}$ – 9 – 2013

$R = (A\ B\ C\ D\ E\ F\ G\ H)$

FDs = { CH → G
A → BC
B → CFH
E → A
F → EG }

How many Candidate keys does the relation have?

→ $(D\_)^+ = \{A\ B\ C\ D\ E\ F\ G\ H\}$

So, here, 4 - ~~Cat~~ candidate keys possible. (AD)(BD)(ED)(FD).

① ✗ $(D)^+ = \{D\}$

② ✓ $(AD)^+ = \{A, D, B, C, F, H, G, E\}$
✓ $(BD)^+ = \{B, D, C, F, H, E, G, A\}$
✗ $(CD)^+ = \{C, D\}$
✓ $(ED)^+ = \{E, D, A, B, C, F, H, G\}$
✓ $(FD)^+ = \{F, D, E, G, A, B, C, H\}$
✗ $(GD)^+ = \{G, D\}$
✗ $(HD)^+ = \{H, D\}$

③

(CD) (GD) (HD) not add (A B E F) add (C G H)

~~ACD~~
~~BCD~~
~~ECD~~
~~FCD~~

{ ✗ $(GCD)^+ = \{G, C, D\}$
✗ $(HCD)^+ = \{H\ C\ D\ G\}$

✗ $(HGD)^+ = \{H, G, D\}$

~~HGD~~

④ ✗ $(DCGH)^+ = \{D\ C\ G\ H\}$

Example - 1 - Candidate Key.

① $R = (ABCDE)$

$FDs = \{AB \to C,\ C \to D,\ B \to E\}$
find candidate key.

→ $(AB)^+ = (A\ B)\ C\ D\ E$
→ candidate key. | Super key $= 2^3 = 8$

② $R = (ABCDE)$

$FD = \{AB \to C,\ C \to D,\ B \to EA\}$
Candidate key = ?

→ $(\ \ )^+ = (A\ B\ C\ D\ E)$
↓
B

① $(B)^+ = \{B, E, A, C, D\}$
B is the CK.

② ~~$(AB)^+ = \{A\ B\ C\ D\ E\}$~~
~~$(CB)^+ = \{C\ B\ E\ A\ D\}$~~
~~$(DB)^+ = \{D\ B\ E\ A\ C\}$~~
~~$(EB)^+ = \{E\ B\ A\ C\ D\}$~~
no. of SK $= 2^4 = 16$

③ $R = ABCDEF$
$FD = \{A \to B,\ C \to D,\ E \to F\}$

→ $(\ \ )^+ = (A\ B\ C\ D\ E\ F)$ CK = ACE
↓
ACE no. of SK $= 2^3 = 8$

$(ACE)^+ = \{A\ C\ E\ B\ D\ F\}$

(4) $R = ABCD$

$FD = \{AB \to CD,\ D \to A\}$

$CK = ?$

$\rightarrow$ $(\ \ )^+ = (ABCD)$
B

$\times\ (B)^+ = B$

$\checkmark\ (AB)^+ = (ABCD)$

$\times\ (CB)^+ = (CB)$

$\checkmark\ (DB)^+ = (DBAC)$

$CK = 2\ (AB, DB)$ — SK_1, SK_2

$SK = SK_1 + SK_2 - (SK_1 \cap SK_2)$

$= 2^2 + 2^2 - 2^1$

$= 8 - 2$

$= 6$

CB

(5) $R = (ABCD)$

$FD = \{AB \to CD,\ C \to A,\ D \to B\}$

$\rightarrow$ $(\ \ )^+ = (ABCD)$

attribute (1)

$A^+ = (A)$

$B^+ = (B)$

$C^+ = (AC)$

$D^+ = (DB)$

Att → (2)

$\checkmark\ AB^+ = ABCD$

$\times\ AC^+ = AC$

$\checkmark\ AD^+ = ADBC$

$\checkmark\ BC^+ = BCAD$

$\times\ BD^+ = BD$

$\checkmark\ CD^+ = CDAB$

CK: $(AB)(AD)(BC)(CD)$

with (3) - attribute -

$\cancel{(AC)}\ \cancel{(BD)}$

(ABCD)

4 - CK is possible (AB, AD, BC, CD)

(6) $R = ABCDEF$

$FD = \{AB \to C,\ C \to D,\ D \to E,\ E \to F,\ F \to A\}$

$\Rightarrow$

$(\underset{B}{\downarrow})^+ = (ABCDEF)$

(1) $\times\ (B)^+ = \{B\}$

(2) $\checkmark\ (AB)^+ = (ABCDEF)$
$\checkmark\ (CB)^+ = (CBDEFA)$
$\checkmark\ (DB)^+ = (DBEFAC)$
$\checkmark\ (EB)^+ = (EBFACD)$
$\checkmark\ (FB)^+ = (FBACDE)$

CK = 5. (AB, CB, DB, EB, FB)

(7)

$R = ABCDEF$

$FD = \{AB \to C,\ C \to D,\ D \to EB,\ E \to F,\ F \to A\}$

$\to$ $(\ \ )^+ = (ABCDEF)$

(1)	(2)
$A^+ = A$	$\checkmark\ (AB)^+ = (ABCDEF)$
$B^+ = B$	$(AE)^+ = (AEF)$
$\checkmark\ C^+ = (CDEBFA)$	$(AF)^+ = (AF)$
$\checkmark\ D^+ = (DEBFAC)$	$\checkmark\ (BE)^+ = (BEFACD)$
$E^+ = (EFA)$	$\checkmark\ (BF)^+ = (BFACDE)$
$F^+ = (FA)$	
(3) $(FAE)^+ = (AEF)$	

CK = 5 (C, D, AB, BF, BE)

(8) R(ABCDEF)

FD = A → ~~ABC~~ BCDEF,
BC → ADEF.
DEF → ABC.

→ $(\quad)^+ = (ABCDEF)$

(1)
✓ $A^+ = (ABCDEF)$
× $B^+ = (B)$
× $C^+ = (C)$
× $D^+ = (D)$
× $E^+ = (E)$
× $F^+ = (F)$

(2)
✓ $BC^+ = (BCADEF)$
$BD^+ = ×$
$BE^+ = ×$
$BF^+ = ×$
$CD^+ = ×$
$CE^+ = ×$
$CF^+ = ×$
$DE^+ = ×$
$DF^+ = ×$
$EF^+ = ×$

(3)
✓ $(DEF)^+ = (DEFABC)$

CK = (A, BC, DEF)

(9) R = (ABCDE)
FD = {A → B, B → C, C → D, D → A}

→ $(\underset{E}{\downarrow})^+ = (ABCDE)$

× $E^+ = \{E\}$
✓ $AE^+ = (AEBCD)$
✓ $BE^+ = (BECDA)$
✓ $CE^+ = (CEDAB)$
✓ $DE^+ = (DEABC)$

CK = 4

(10)

$R = (ABCDE)$

FD: $A \rightarrow BC$

$CD \rightarrow E$

$B \rightarrow D$

$E \rightarrow A$

$\rightarrow$

$(\ )^+ = (ABCDE)$

(1)

✓ $A^+ = (ABCDE)$

$\times \begin{cases} B^+ = (BD) \\ C^+ = (C) \\ D^+ = (D) \end{cases}$

✓ $E^+ = (EABCD)$

(2)

✓ $(BC)^+ = (BCDEA)$

$(BD)^+ = (BD)$

✓ $(CD)^+ = (CDAEB)$

(3) ~~CBD=~~

$CK = 4\ (A, E, BC, CD)$

Example - candidate key for sub relation -

(1) $R(ABCD)$

$\{AB \rightarrow CD,\ D \rightarrow A\}$

what are the canditate keys of sub relation

$R_1(BCD)$

$\rightarrow$ $B^+ = B$

$C^+ = C$

$D^+ = DA$

this table is not going to have any CK with one attribute.

$BC^+ = BC$
$CD^+ = CDA$
✓ $BD^+ = BDAC$

here candidate key = BD.

(2) R (ABCDEF)
{ AB → C , B → D, AD → F, C → D, D → E, E → F, E → D }
Find candidate keys for R_1(DEF).

→ ✓ $D^+ = DEF$
✓ $E^+ = EDF$
✗ $F^+ = F$

For this table CK = D and E.

(3) R (ABCDE)
{ A → BC , CD → E , B → D, E → A }
What are the CK of R_1 (ABCE)

→ ✓ $A^+ = BCDEA$	✓ ~~$BC^+ = BCDEA$~~
✗ $B^+ = BD$	✗ ~~$BD^+ = BD$~~
✗ $C^+ = C$	✓ ~~$CD^+ = CDEAB$~~
✓ $E^+ = EABC$	✓ $BC^+ = BCDEA$

for relation R_1(ABCE) CK = (A, E, BC)

Example: Checking additional FDs —

(1)

R(AB)

X → Y ∅ → ∅

∅, A, B, AB ∅, A, B, AB

total possible FDs = 16 3+1+1

✓ ∅ → ∅	✓ A → ∅	✓ B → ∅	✓ AB → ∅
✗ ∅ → A	✓ A → A	✓ B → A	✓ AB → A
✗ ∅ → B	✓ A → B	✓ B → B	✓ AB → B
✗ ∅ → AB	✓ A → AB	✓ B → AB	✓ AB → AB

(i)

$\boxed{\begin{array}{l} A \to B \\ B \to A \end{array}}$ ← 13-FDs are implied by

(ii)

$\boxed{A \to B}$ ← 11-FDs are implied by

(11-FDs valid according to FDs)

→ 10 additional we can derive from (A→B).

(ex-2)

R(ABC)

FDS : {A → B , B → C}

What are all ~~Es~~ additional FDs that can derive from given FDs.

→

$X \rightarrow Y$
$\downarrow \quad \downarrow$
$2^3 \quad 2^3$

$8 \times 8 = 64$

$\emptyset \rightarrow \emptyset$ – ①
$A \rightarrow \{ABC\}$ – $2^3 = 8$
$B \rightarrow \{BC\}$ – $2^2 = 4$
$C \rightarrow \{C\}$ – $2^1 = 2$
$AB \rightarrow \{ABC\}$ – $2^3 = 8$
$BC \rightarrow \{BC\}$ – $2^2 = 4$
$AC \rightarrow \{ABC\}$ – $2^3 = 8$
$ABC \rightarrow \{ABC\}$ – $2^3 = 8$

$A^+ = ABC$
$A \rightarrow \emptyset$
$\rightarrow A$
$\rightarrow B$
$\rightarrow C$
$\rightarrow AB$
$\rightarrow BC$
$\rightarrow CA$

$C^+ = C$
$\rightarrow \emptyset$

$B^+ = BC$
$\rightarrow \emptyset$
$\rightarrow B$
$\rightarrow C$
~~ee~~ BC

43 – Functional Dependencies possible with given FDs.

$R(A_1, A_2, A_3 \ldots A_n)$

$X \rightarrow Y$
$2^n \quad 2^n$

$2^n \times 2^n = 2^{2n}$ – total FDs possible with 'n' attributes

Question - g - 2005

In a schema with attributes A, B, C, D and E, following set of FD's are given
$A \to B$; $A \to C$, $CD \to E$, $B \to D$, $E \to A$.
Which of the following FDs is Not Implied by the above set ?

✓ a) $CD \to AC$ ✗ b) $BD \to CD$
✓ c) $BC \to CD$ ✓ d) $AC \to BC$

→ $CD^+ = CDEA$ $AC^+ = ACBD$

$BC^+ = BCDEA$ $BD^+ = BD$

- Equivalence of FDs :

① F: { $A \to C$, $AC \to D$, $E \to AD$, $E \to H$ }
G: { $A \to CD$, $E \to AH$ }

→ $\boxed{F \supseteq G}$ ✓
→ $\boxed{G \supseteq F}$ ✓ $\boxed{F \cong G}$

$A^+ = ACD$	$A^+ \to ACD$
$E^+ = EADH$	$E \to EAHCD$

So, both the FDs are equivalent.
$F \cong G$

(2) F: {A → B, ~~B → C~~, C → D}
G: {A → BC, C → D}

→ ✓ $\boxed{F \supseteq G}$
× $\boxed{G \supseteq F}$ → If both are satisfy then we can say F ≅ G.

A → ABCD	A → BCDA
	~~B~~ C → D

F and G are not equivalent.

(3) F: {A → B, AB → C, D → AC, ~~D → E~~}
G: {A → BC, D → AB}

→ ✓ $\boxed{F \supseteq G}$ | ✗ $\boxed{G \supseteq F}$

A → ABC	A → ABC
D → DACB	D → ~~A~~ DABC

~~F and G are not~~ F ≇ G not equivalent.

(4) F: {A → B, B → C, C → A}
G: {A → BC, B → A, C → A}

→ ✓ $\boxed{F \supseteq G}$ | ✓ $\boxed{G \supseteq F}$

A → ~~BC~~ ABC	A → ~~A~~ ABC
B → BCA	B → BAC
C → ABC	C → ABC

F ≅ G

- Minimise FDs set:

F
⇓ minimal
Gc

Procedure to find minimal set:

1) Split the FDs such that RHS contain single attribute.
   Ex: $A \to BC \Rightarrow A \to B$ and $A \to C$

2) Find the redundant FDs and delete them from the set.
   Ex: $\{A \to B,\ B \to C,\ \cancel{A \to C}\}$
   $\Rightarrow \{A \to B,\ B \to C\}$

3) Find the redundant attributes on LHS and delete them.
   Ex: $AB \to C$, A — can be deleted if B^+ — contains 'A'.

example —

① FD: $\{A \to C,\ AC \to D,\ E \to AD,\ E \to H\}$

→ ② $A \to C,\ AC \to D,\ E \to A,\ \cancel{E \to D},\ E \to H$

$A^+ = A$
$(AC)^+ = AC$

$E^+ = EACD$
$E^+ = ACD$

③ $A \to C,\ A\cancel{C} \to D,\ E \to A,\ E \to H$
$C^+ = C$
$A^+ = AC$

$A \to C,\ A \to D,\ E \to A,\ E \to H$ — minimal set

$A \to CD,\ E \to AH$

(2) F: $\{A \rightarrow B, C \rightarrow B, D \rightarrow ABC, AC \rightarrow D\}$

→ 1) $A \rightarrow B$, $C \rightarrow B$, $D \rightarrow A$, $D \rightarrow B$, $D \rightarrow C$, $AC \rightarrow D$

✓ ✓ ✓ ✗ ✓ ✓

$A^+ = A$ $\quad C^+ = C$ $\quad D^+ = ADB$

$D^+ = DBC$ $\quad D^+ = ADCB$

$(AC)^+ = ACB$

2) $A \rightarrow B, C \rightarrow B, D \rightarrow A, D \rightarrow C, AC \rightarrow D$

→ minimal set of FDs. $\quad C^+ = CB$

$A^+ = BA$

(3) FD: $\{AB \rightarrow C, D \rightarrow E, E \rightarrow C\}$ is the minimal cover of

$\{AB \rightarrow C, D \rightarrow E, AB \rightarrow E, E \rightarrow C\}$

→ 1st check both are equivalent or not.

$(AB)^+ \rightarrow ABC$	$AB^+ \rightarrow ABCE$
$D^+ \rightarrow DEC$	$D \rightarrow E$
$E^+ \rightarrow EC$	$E \rightarrow C$

both are not equivalent.

Lossless decomposition =

ABC

A	B	C
a_1	b_1	c_1
a_2	b_1	c_1
a_1	b_2	c_2

AB		BC	
a_1	b_1	b_1	c_1
a_2	b_1	b_2	c_2
a_1	b_2		

A	B	C
a_1	b_1	c_1
a_2	b_1	c_1
a_1	b_2	c_2

Lossless decomposition

AB		AC	
a_1	b_1	a_1	c_1
a_2	b_1	a_2	c_1
a_1	b_2	a_1	c_2

A	B	C
a_1	b_1	c_1
× a_1	b_1	c_2
a_2	b_1	c_1
× a_1	b_2	c_1
a_1	b_2	c_2

~~lossless~~ Lossy decomposition

A	BC	
a_1	b_1	c_1
a_2	b_2	c_2

A	B	C
a_1	b_1	c_1
a_1	b_2	c_2
a_2	b_1	c_1
× a_2	b_2	c_2

When extra tuple added thats call data loss.

→ Every Decomposition is not lossless decomposition
→ Some of Decomposition is lossless decomposition

→ We need lossless decomposition.
→ Key is one of the table decomposition not be losse.

→ When devide the table see the common attribute

$$R \to R_1,\ R_2$$

$R_1 \bowtie R_2 \supset R$ − lossy
$R_1 \bowtie R_2 = R$ − lossless

$$(R_1 \cap R_2) \to R_1 \quad = (R_1 - R_2)$$

or

$$(R_1 \cap R_2) \to R_2 \quad = (R_2 - R_1)$$

TO DOWNLOAD THE COMPLETE PDF

CLICK ON THE LINK GIVEN BELOW

WWW.GATENOES.IN

GATE CSE NOTES