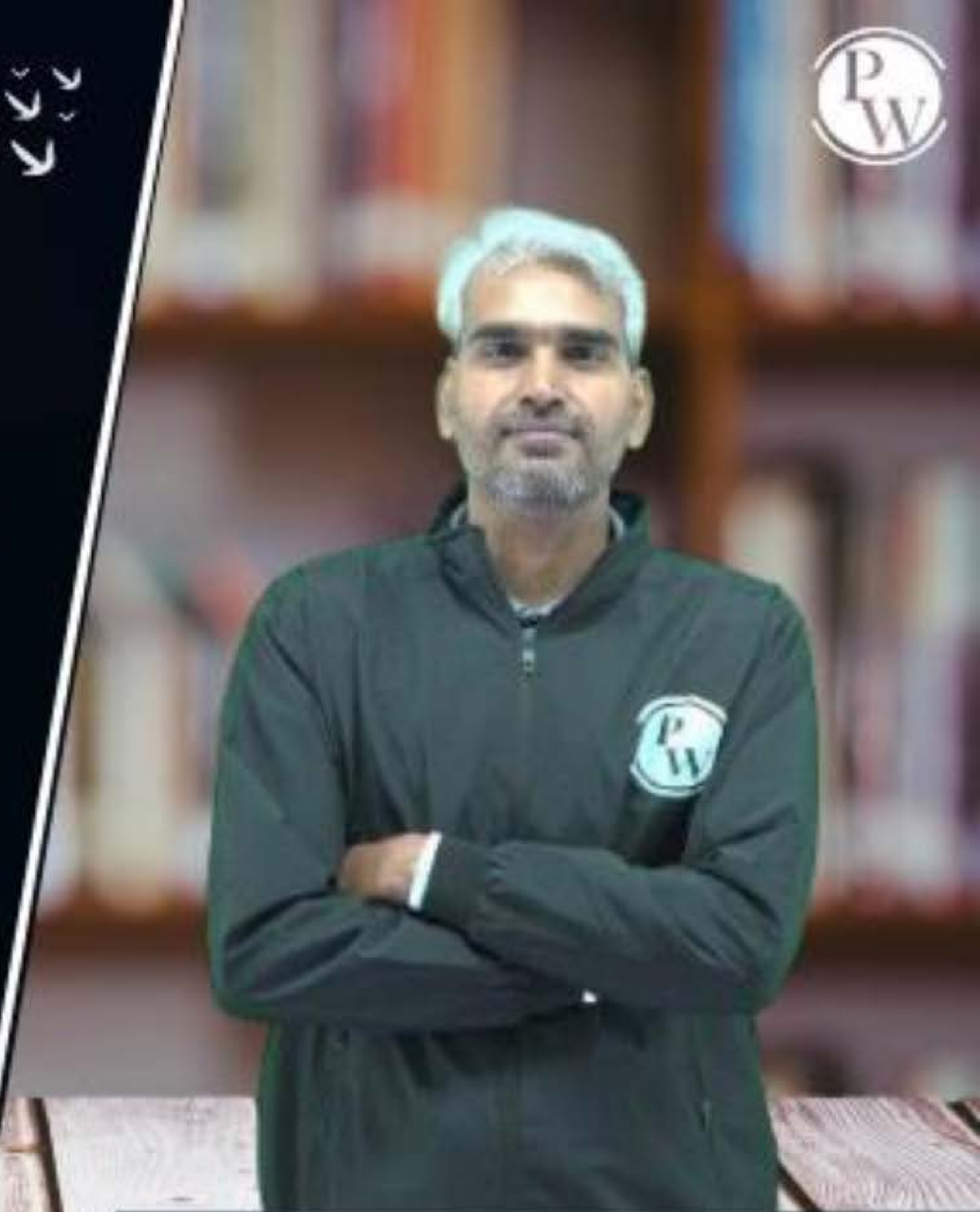
Computer Science & IT

Database Management System

Relational Model & Normal Forms

Lecture No. 09



By- Vishal Sir

Recap of Previous Lecture

- **Topic** Properties of decomposition
- **Topic** Dependency preserving decomposition

Topics to be Covered

Topic — Lossless join decomposition

Topic — Normal forms

H.W. Q :- Let $R(A, B, C, D, E, F)$ be the relational schema with FD set $F = \{AB \to C,\ BC \to A,\ AC \to B,\ B \to D,\ AD \to E,\ E \to F\}$

Which of the following decomposition of R is dependency preserving decomposition.

(1) $D_1 = \{R_1(ABC),\ R_2(ACDF),\ R_3(EF)\}$

(2) $D_2 = \{R_1(ABC),\ R_2(ABDE),\ R_3(EF)\}$

$R(ABCDEF)$

$AB \to C,\ BC \to A,\ AC \to B,$
$B \to D,\ AD \to E,\ E \to F$

$R_1(ABC) \qquad R_2(ACDF) \qquad R_3(EF)$

$$F_1 = \left\{ \begin{array}{l} AB \to C \\ BC \to A \\ AC \to B \end{array} \right\} \qquad F_2 = \left\{ \begin{array}{l} AC \to DF \\ AD \to F \end{array} \right\} \qquad F_3 = \{ E \to F \}$$

$(B)^+$ w.r.t $F_1 \cup F_2 \cup F_3 = \{B\}$ $\therefore$ $B \to D$ is lost

$(AD)^+$ w.r.t $F_1 \cup F_2 \cup F_3 = \{A, D, F\}$ $\therefore$ $AD \to E$ is lost

$\therefore$ Not a dep. preserving decomposition

$AB \to C,\ BC \to A,\ AC \to B,$
$B \to D,\ AD \to E,\ E \to F$

$R(ABCDEF)$

$R_1(ABC)$ $\qquad$ $R_2(ABDE)$ $\qquad$ $R_3(EF)$

$$F_1 = \begin{Bmatrix} AB \to C \\ BC \to A \\ AC \to B \end{Bmatrix} \qquad F_2 = \begin{Bmatrix} B \to D \\ AD \to E \\ AB \to DE \end{Bmatrix} \qquad F_3 = \{E \to F\}$$

All FDs are preserved in $F_1 \cup F_2 \cup F_3$

$\therefore$ Dep. preserving decomposition

Lossless Join decomposition

Topic : Lossless Join decomposition

If we decompose a relation R with FD set F into sub-relations R1, R2,.....,Rn with FD sets F1, F2,.....,Fn respectively, then for this decomposition to be called lossless join decomposition following property must hold true.

$$\boxed{R1 \bowtie R2 \bowtie \ldots \bowtie Rn = R}$$

Topic : Lossless Join decomposition

Let relation R is decomposed into sub-relations R1, R2,.....,Rn

In general, $R1 \bowtie R2 \bowtie \ldots \bowtie Rn \supseteq R$

if, $R1 \bowtie R2 \bowtie \ldots \bowtie Rn = R$ then, Lossless join decomposition

if, $R1 \bowtie R2 \bowtie \ldots \bowtie Rn \supset R$ then, Lossy join decomposition

$R1 \bowtie R2 \bowtie \ldots \bowtie Rn \subset R$ (not possible)

Topic : Natural Join (⋈)

- Natural Join(⋈) is a derived Relational Algebra operation, which is derived using three basic Relational Algebra operation

  - Projection (π)
  - Selection(σ)
  - Cross Product ($\times$)

Topic : Projection (π)

It is used to project the column data from a relation based on the attributes specified with projection operation.

eg: Consider the following relational schema, $R(A_1, A_2, A_3, A_4, A_5)$

Syntax:

$\pi_{\text{List of attributes required in o/p}}(\text{Relation\_name})$

Attributes required in o/p

Name of relⁿ

$$\underbrace{\pi_{A_1, A_3, A_4}(R)}$$

Resulting schema will Contain Only three attributes, i.e:, A_1, A_3 & A_4

eg:

Enroll

Sid	Cid	Branch
S_1	C_1	CS
S_1	C_2	CS
S_2	C_2	IT
S_3	C_3	CS

$$\underbrace{\pi_{Sid,Cid,Branch}(Enroll)}_{\text{O/p of this R.A. Expression will be the Complete "Enroll" table}} \equiv (Enroll)$$

If we do not use the projection opn, then O/p will contain all attributes of relational schema

+ Retrieve the Sids of the students who enrolled for some Courses.

then $\pi_{Sid}(Enroll) \Rightarrow$ O/p:

~~Sid~~
~~S_1~~
~~S_1~~
~~S_2~~
~~S_3~~

Sid
S_1
S_2
S_3

Note: Relational Algebra query will always produce distinct tuples

eg:

Enroll

Sid	Cid	Branch
S_1	C_1	CS
S_1	C_2	CS
S_2	C_2	IT
S_3	C_3	CS

$\pi_{Sid,\, Cid}(\text{Enroll}) \Rightarrow$ o/p =

Sid	Cid
S_1	C_1
S_1	C_2
S_2	C_2
S_3	C_3

tuplewise they are distinct

Topic : Selection(σ)

- It is used to select the tuples(records) from underlying relation based on the predicate condition specified with selection operation.

Syntax: $\sigma_{\text{Cond}^n \text{ to select tuples}}$ (Relation_name)

op^n

If we do not specify selection Cond^n then all the tuples will be selected

eg:

Enroll

Sid	Cid	Branch
S_1	C_1	CS
S_1	C_2	CS
S_2	C_2	IT
S_3	C_3	CS

Retrieve the records of the students who enrolled for Course with Cid = 'C_1'

i.e all attributes

$$\sigma_{Cid = 'C_1'}(Enroll) \Rightarrow o/p =$$

Sid	Cid	Branch
S_1	C_1	CS

No projection opn

∴ all attributes will be present

OR

$$\pi_{Sid}\left(\sigma_{Cid='C_1' \vee Branch='CS'}(Enroll)\right) \Rightarrow o/p =$$

Sid
S_1
S_3

eg:

Enroll

Sid	Cid	Branch
S_1	C_1	CS
S_1	C_2	CS
S_2	C_2	IT
S_3	C_3	CS

$$\pi_{Sid}\left(\sigma_{Cid='C_1' \wedge Branch='CS'}(Enroll)\right) \Rightarrow o/p =$$

Sid
S_1

Topic : Cross Product (×)

/ Cross Join / Cartesian Product / Cartesian Join

- Cross-product is a binary operation. Let R and S are any two relation, then cross product 'R × S' will result in all attributes of R followed by all attribute of S with all possible combinations of tuples from R and S.

  i.e. Every tuple of R will join with Every tuple of S 'x' attributes

R

'm' tuples

A	B	C
1	2	3
1	3	5
3	6	9

'y' attributes

S

'n' tuples

B	D
2	7
3	8
3	5

"x+y" attributes

R x S =

"m.n" tuples

R.A	R.B	R.C	S.B	S.D
1	2	3	2	7
1	2	3	3	8
1	2	3	3	5
1	3	5	2	7
1	3	5	3	8
1	3	5	3	5
3	6	9	2	7
3	6	9	3	8
3	6	9	3	5

Topic : Natural Join (⋈)

Natural join(⋈) is a derived relational algebra operation which is derived using cross product, selection and projection as follows:

Let R and S are any two relations then,

R⋈S = Step-1: Obtain "R × S"

R natural S join

Step-2: Select the tuples from "R × S" based on the equality condition on all common attributes of R and S.

Step-3: Project distinct attributes from the result of step-2.

R

A	B	C
1	2	3
1	3	5
3	6	9

S

B	D
2	7
3	8
3	5

$$R \bowtie S = \pi_{R.A,\, R.B,\, R.C,\, S.D}\left(\sigma_{R.B = S.B}\,(R \times S)\right)$$

Common attribute will be projected only once

$R \times S =$

R.A	R.B	R.C	S.B	S.D
1	2	3	2	7
1	2	3	3	8
1	2	3	3	5
1	3	5	2	7
1	3	5	3	8
1	3	5	3	5
3	6	9	2	7
3	6	9	3	8
3	6	9	3	5

$R \bowtie S =$

A	B	C	D
1	2	3	7
1	3	5	8
1	3	5	5

Note:- Let $R(A,B,C)$ & $S(B,C,D)$

$$R \bowtie S = \pi_{R.A,\,R.B,\,R.C,\,S.D}\left(\sigma_{(R.B = S.B \,\wedge\, R.C = S.C)}(R \times S)\right)$$

Equality Cond^n on all Common attributes b/w R & S

Note: Let $R(A,B,C)$ & $S(D,E)$ {i.e. No common attributes b/w R & S}

$$R \bowtie S = \pi_{\underbrace{R.A,R.B,R.C,S.D,S.E}_{\text{Project all attributes of } R \times S}}(R \times S) \therefore \equiv R \times S$$

No Common attributes, ∴ No Selection Condⁿ.

Hence, all the tuples of R X S will be Selected

i.e. Select all tuples of R×S & Project all attributes of R×S

Note:

If there are no common attributes b/w R & S, then o/p of '$R \bowtie S$' will be exactly same as '$R \times S$'

eg: let

R =

A	B
1	3
2	5
6	9

S =

B	C
2	8
4	6
6	9

What will be o/p of $R \bowtie S$ =

A	B	C

there will be no tuple in the o/p

i.e, o/p of $R \bowtie S$ will be Empty

Lossless Natural Join.

HW

Consider the following relational table.

R

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

Case ① If relation R is decomposed into two subrelations $R_1(AB)$ & $R_2(BC)$, then check whether the decomposition is lossless Join decomposition or not?

$R_1 =$

A	B
1	1
2	1
3	2

$R_2 =$

B	C
1	1
1	2
2	1

$R_1 \bowtie R_2 =$

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

tuples from R

Extra tuples (Spurious tuples)

- Common attribute b/w R_1 & R_2 is 'B'
- & Values of B are neither unique in R_1 nor unique in R_2

& decomposition is Lossy Join

$R_1 \bowtie R_2 \supset R$ ∴ Lossy Join decomposition

H.W.

Consider the following relational table.

R

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

• Common attribute b/w R_1 & R_2 is A and values of A are unique in both the relations, and Join is lossless

② If relation R is decomposed into two subrelations $R_1(AB)$ & $R_2(AC)$, then check whether the decomposition is lossless Join decomposition or not?

R_1 =

A	B
1	1
2	1
3	2

R_2 =

A	C
1	1
2	2
3	1

$R_1 \bowtie R_2$ =

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

$R_1 \bowtie R_2 = R$

$\therefore$ lossless Join decomposition

H.W. Consider the following relational table.

R

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

③ If relation R is decomposed into two subrelations $R_1(AB)$ & $R_2(BC)$, then check whether the decomposition is lossless Join decomposition or not?

$R_1 =$

A	B
1	1
2	1
3	2

$R_2 =$

B	C
1	2
2	1

$R_1 \bowtie R_2 =$

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

→ Common attribute b/w R_1 & R_2 is 'B'

Values of B are not unique in R_1, but values of B are Unique in R_2 and Join is lossless.

$R_1 \bowtie R_2 = R$

∴ Lossless Join decomposition

Topic : NOTE

* If Relation R is decomposed into two subrelations R_1 & R_2, then this decomposition is lossless join decomposition if and only if following three conditions holds true.

(1) Attributes of R_1 $\cup$ Attributes of R_2 = All attributes of R

(2) Attributes of R_1 $\cap$ Attributes of R_2 $\neq \emptyset$

(3) Common attribute b/w R_1 & R_2 must be a Super key of at least on R_1 or R_2

#Q. Let R (A, B, C, D, E) be the relational schema with following FD set

$F = \{ AB \to C, C \to D, B \to E)$

Which of the following decomposition is/are lossless join decomposition?

(i){R1(ABC), R2(CD)} Att of R_1 $\cup$ Attribute of R_2 $\neq$ Attribute of R $\therefore$ Lossy

(ii){R1(ABC), R2(DE)} Attribute of R_1 $\cap$ Attribute of R_2 = ϕ $\therefore$ Lossy

(iii){R1(ABC), R2(CDE)}
① All attributes present
② Common attribute is C
③ $(C)^+ = \{C, D\}$, i.e. C is neither a S.K of R_1 nor a S.K. of R_2 $\therefore$ Lossy

(iv){R1(ABCD), R2(BE)}
① All attributes present
② Common attribute is B
③ $(B)^+ = \{B, E\}$ all attributes of R_2 $\therefore$ SK of R_2

All three Cond^n satisfied. $\therefore$ lossless Join

Note :- ① $R_1 \bowtie R_2 = R_2 \bowtie R_1$ (Datawise)

{i.e. Natural join is Commutative}

② $(R_1 \bowtie R_2) \bowtie R_3 = R_1 \bowtie (R_2 \bowtie R_3)$ (Datawise)

{i.e. Natural join is associative}

#Q. Let R (A, B, C, D, E, F) be the relational schema with following FD set

$F = \{ AB \rightarrow C, BC \rightarrow A, AC \rightarrow B, AD \rightarrow E, B \rightarrow D, E \rightarrow F)$

Which of the following decomposition is/are lossless join decomposition.

(i) {R1(ABC), R2(ADF), R3(ACDE)}

(ii) {R1(ABC), R2(ABDE), R3(EF)}

(iii){R1(AB), R2(BC), R3(ABDE), R4(EF)}

(i) $R_1(ABC)$ $R_2(ADF)$ $R_3(ACDE)$

$\cap = A$

$(A)^+ = \{A\}$

i.e. not a S.K. of any relation

∴ Don't join in this order

$\cap = AC$

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

S.K of both the relⁿ

∴ We can join R_1 & R_3

$(R_1 \bowtie R_3)(ABCDE)$

$\cap = AD$

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

∴ S.K of R_2

∴ We can join $(R_1 \bowtie R_3)$ with R_2

$((R_1 \bowtie R_3) \bowtie R_2)(ABCDEF)$

If there exist any order in which relations can be joined such that join is lossless at every point of join, then overall decomposition is lossless join decomposition

└ If there exists no such order, then decomposition is lossy

(ii) $R_1(ABC)$ $R_2(ABDE)$ $R_3(EF)$

$\cap = AB$

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

S.K. of both the relations

$\therefore$ We can join

$(R_1 \bowtie R_2)(ABCDE)$

$\cap = E$, $(E)^+ = \{E, F\}$

$\therefore$ S.K. of R_3

$\therefore$ We can join

$((R_1 \bowtie R_2) \bowtie R_3)(ABCDEF)$

#Q. Let R (A, B, C, D, E, F) be the relational schema with following FD set

$F = \{ AB \rightarrow C, BC \rightarrow A, AC \rightarrow B, AD \rightarrow E, B \rightarrow D, E \rightarrow F)$

Which of the following decomposition is/are lossless join decomposition.

(i) {R1(ABC), R2(ADF), R3(ACDE)}

(ii) {R1(ABC), R2(ABDE), R3(EF)}

(iii){R1(AB), R2(BC), R3(ABDE), R4(EF)} (it is Lossy)

Topic : Normalization

- Normalization is the process of decomposing the relation into sub-relations, such that redundancy is reduced or eliminated.

Topic : Normal forms

There are various normal forms

1 NF	2 NF	3 NF	BCNF	4 NF

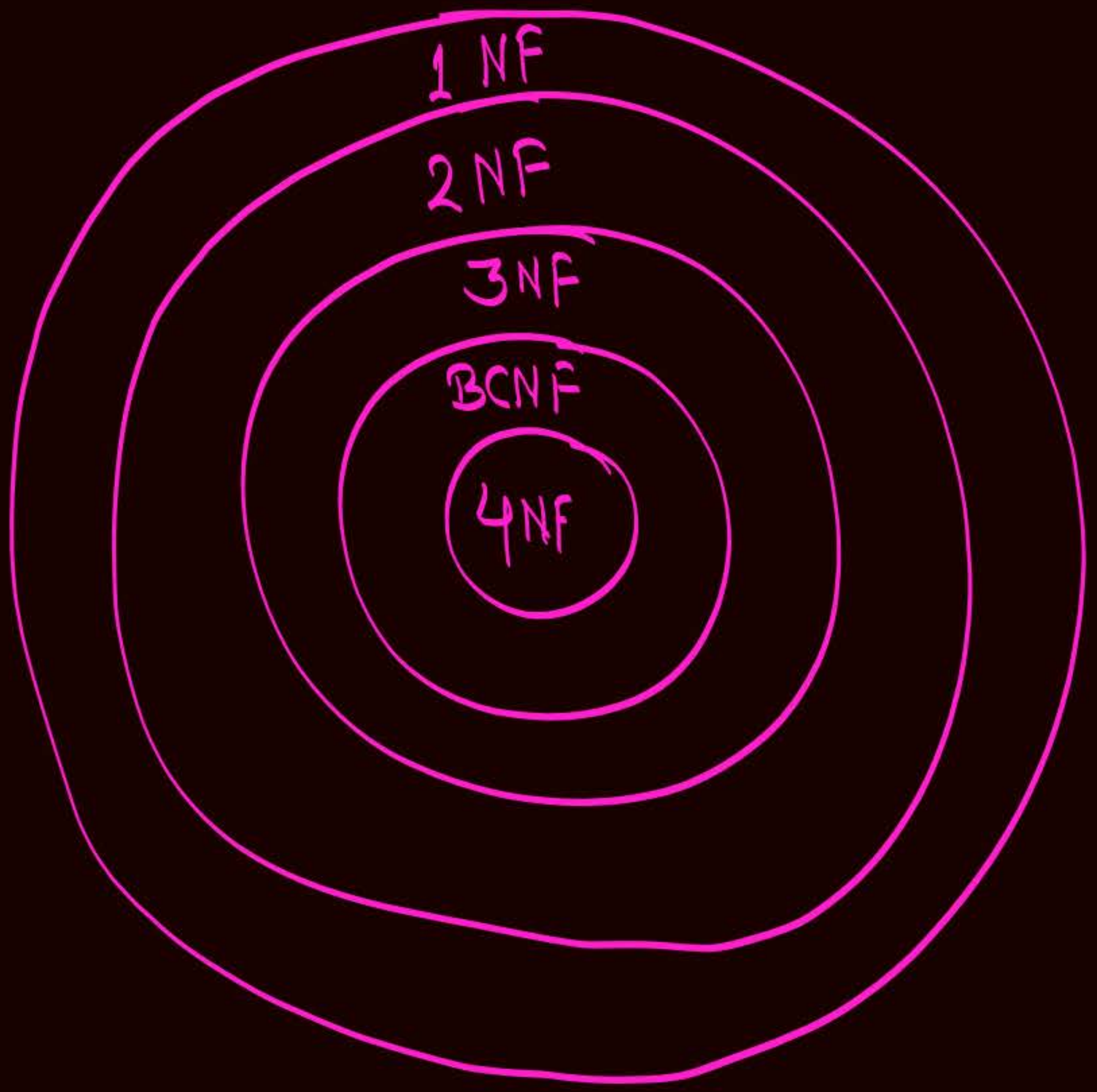
- Upto BCNF we try to eliminate the redundancy present in the relation because of functional dependencies.
- If relation is in BCNF, then there will be no redundancy in that relation because of functional dependencies, but a relation in BCNF may still suffer from redundancies present in it because of multivalued dependency.

- 4 NF is related to multi-valued dependency.
- In 4NF we try to eliminate the redundancy present in the relation because of multi-valued dependency.

* Every relation which is in 2NF, is also in 1NF.
* Every relation which is in 3NF, is also in 2NF and hence also in 1NF.

⋮

and so on



Topic : First normal form (1NF)

For a database to be in "1NF" it must not contain any muti-valued attribute { i.e. all attributes must be simple and single (atomic) valued }

eg:

Multi-valued attribute

Sid	Courses
S_1	$\{C_1, C_2\}$
S_2	$\{C_2, C_3\}$
S_3	C_3

it is not a relation

Multi-valued attribute are Present ∴ it is not in "1NF"

Convert multi-valued attribute in to single valued attribute

Sid	Course
S_1	C_1
S_1	C_2
S_2	C_2
S_2	C_3
S_3	C_3

Now, Course is a single valued attribute

No multi-valued attribute is present ∴ It is at least in "1NF"

* By default normal form of relation is 1NF.

{i.e, Every relation is at least in 1NF}

Topic : Redundancy in relation because of FD

Rule 1 :- In a functional dependency "X → Y", if "X" is a Super Key, then it does not cause any redundancy in the relation

Rule 2 :- In a functional dependency "X → Y" if X is not a super Key, then it may cause redundancy in the relation

Possible types of non-trivial FDs which may cause redundancy in the relation

<u>Note</u>:- In $X \rightarrow Y$, if X is not a Superkey, then Y can never be a Super key.

∴ If $X \rightarrow Y$ causes redundancy in the relation, then neither L.H.S. <u>nor</u> R.H.S. of that FD can be a Super key.

Possible types of non-trivial FDs which may cause redundancy in the relation

$X \longrightarrow Y$

Type ① Proper subset of a C.K $\longrightarrow$ Non-Prime attributes

Type ② (Proper Subset of a Candidate Key + Non-prime attributes) $\longrightarrow$ Non-prime attribute

Type ③ Non-prime attributes $\longrightarrow$ Non-prime attributes

Non-Prime Attributes $\longrightarrow$ P.S.C.K. {Such FDs are not Possible}

Proper subset of a C.K $\longrightarrow$ Proper subset of same C.K Such FD is not Possible

Type ④ Proper subset of one C.K $\longrightarrow$ Proper Subset of some other C.K

Type ⑤ (Proper Subset of one Candidate Key + Non-prime attributes) $\longrightarrow$ Proper Subset of some other C.K

2 mins Summary

Topic Lossless join decomposition

Topic Normal forms

Slide

THANK - YOU

Slide