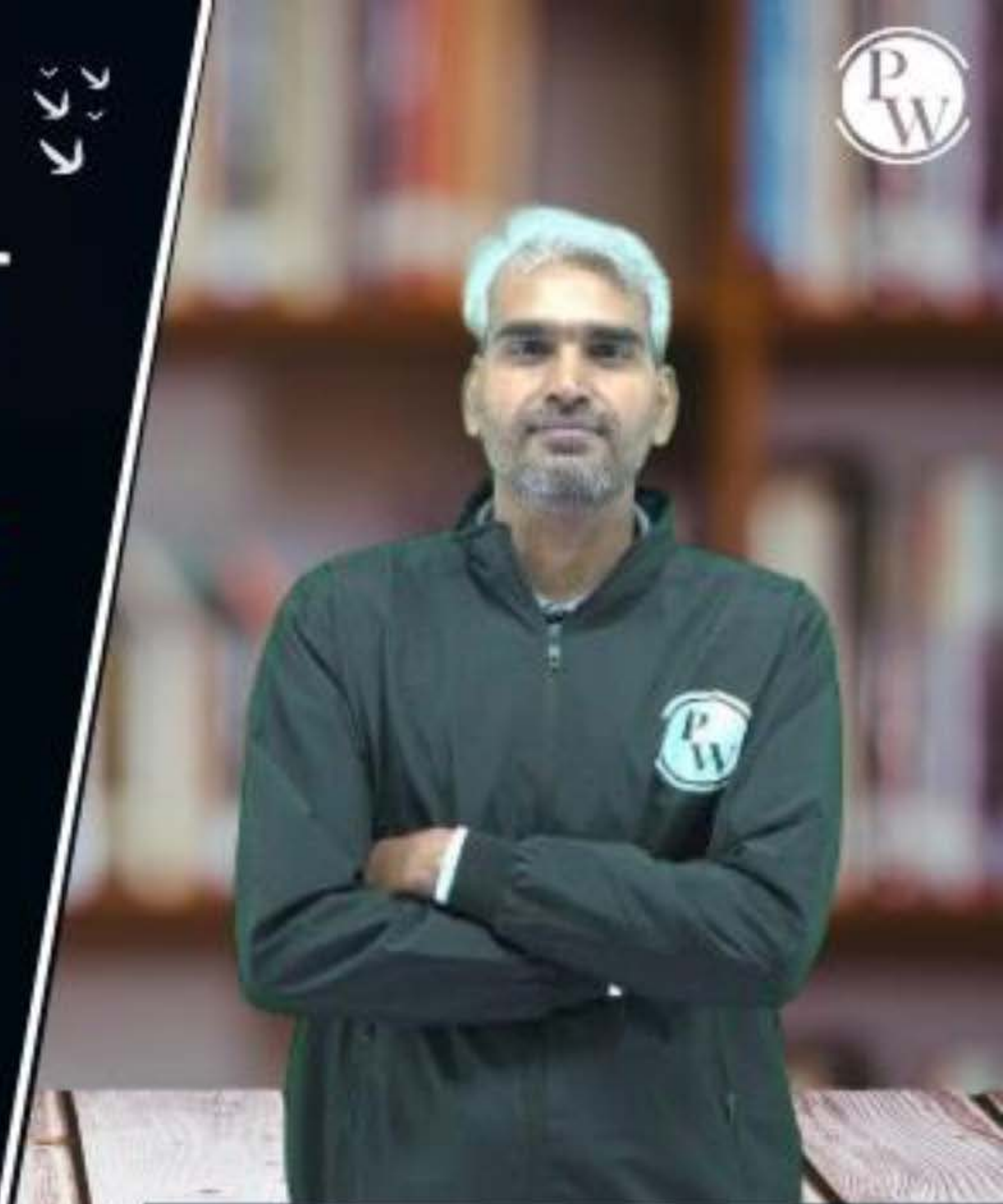
Computer Science & IT

Database Management System

Relational Model & Normal Forms

Lecture No. 08



By- Vishal Sir

Recap of Previous Lecture

Topic Number of superkeys in a relation

Topic Normalization (Schema refinement)

Topics to be Covered

- Topic: Properties of decomposition
- Topic: Dependency preserving decomposition
- Topic: Lossless Join decomposition

Topic : Schema refinement (Normalization)

duplicated data

Normalization is a process of reducing / eliminating the redundancy present in the relational table

Student information

Student_Enrollment_Info

Course information

Sid $\rightarrow$ Sname

Cid $\rightarrow$ Cname

Sid	Sname	Cid	Cname
S_1	A	C_1	DM
S_1	A	C_2	DBMS
S_2	A	C_2	DBMS
S_3	B	C_2	DBMS
S_3	B	C_3	DM

Redundancy

it is information
(not redundancy)

Topic : Redundancy in a relation

If independent informations are stored in the same table, then redundancy is possible

Student_Enrollment_Info

$\underbrace{\text{Sid} \quad \text{Sname}}_{}$ Student information

$\underbrace{\text{Cid} \quad \text{Cname}}_{}$ Course information

$Sid \rightarrow Sname$

$Cid \rightarrow Cname$

Sid	Sname	Cid	Cname
S_1	A	C_1	DM
S_1	A	C_2	DBMS
S_2	A	C_2	DBMS
S_3	B	C_2	DBMS
S_3	B	C_3	DM

it is not redundancy {it is information}

it is redundancy

Topic : Problems because of redundancy

* If redundancy is present in the relation, then various problems are possible

* (1) It requires more storage space { it is not a very big Problem }

Imp
(2) Insertion Anomaly
(3) Deletion Anomaly
(4) Updation Anomaly

Student_Enrollment_Info

Sid	Sname	Cid	Cname
S_1	A	C_1	DM
S_1	A	C_2	DBMS
S_2	A	C_2	DBMS
S_3	B	C_2	DBMS
S_3	B	C_3	DM

Let,

Sid → Sname
&
Cid → Cname

are only two FDs that holds in the relation.

∴ Candidate key of the relation is {Sid, Cid}

If we define a primary key for the relation then {Sid, Cid} will be primary key

If {Sid, Cid} is the Primary key then Sid & Cid are not allowed to take "NULL" values

Insertion Anomaly:-

Student_Enrollment_Info

$Sid \rightarrow Sname$

$Cid \rightarrow Cname$

Attributes of primary key

Sid	Sname	Cid	Cname
S_1	A	C_1	DM
S_1	A	C_2	DBMS
S_2	A	C_2	DBMS
S_3	B	C_2	DBMS
S_3	B	C_3	DM
NULL	NULL	C_4	AI

Try to insert the information of a new Course (C_4, AI)

But it is not allowed because Sid is the attribute of Primary Key.

If no student has Enrolled for this Course, then we must set Sid as NULL

Insertion Anomaly :- If independent informations are stored in the same relational table, then sometimes it may not be possible to insert one information without inserting other independent information.

eg: We can not insert the information of a new Course (C4, AI) until some students enroll for that Course, because "Sid" Can not be NULL.

Deletion Anomaly :-

Suppose we want to delete the information of student with "Sid = S_1"

Student_Enrollment_Info

Sid	Sname	Cid	Cname
~~S_1~~ NULL	~~A~~ NULL	C_1	DM
~~S_1~~ NULL	~~A~~ NULL	C_2	DBMS
S_2	A	C_2	DBMS
S_3	B	C_2	DBMS
S_3	B	C_3	DM

Sid → Sname

Cid → Cname

But it is not allowed because Sid can not be NULL

∴ We will have to delete the Complete tuple w.r.t. Sid = S_1

Deletion Anomaly :-

Suppose we want to delete the information of student with "Sid = S_1"

delete tuples with Sid = S_1

Student_Enrollment_Info

Sid	Sname	Cid	Cname
S_1	A	C_1	DM
S_1	A	C_2	DBMS
S_2	A	C_2	DBMS
S_3	B	C_2	DBMS
S_3	B	C_3	DM

Sid → Sname

Cid → Cname

If we delete all the tuples w.r.t. Sid = S_1, then we lose the information of course with "Cid = C_1"

Deletion Anomaly :- Some times when we try to delete One independent information, we may lose some other independent information.

eg: If we try to delete the information of student "S_1" then information of Course with $Cid = C_1$ is lost.

Updation Anomaly:-

Consider a situation where we want to update the name of Course with "Cid = C_1" to "DB&WH" from "DBMS"

Updation will be required in all duplicate Copies ⟹ ∴ it will be a time consuming operation

Student_Enrollment_Info

Sid	Sname	Cid	Cname
S_1	A	C_1	DM
S_1	A	C_2	~~DBMS~~ DB&WH
S_2	A	C_2	~~DBMS~~ DB&WH
S_3	B	C_2	~~DBMS~~ DB&WH
S_3	B	C_3	DM

$Sid \rightarrow Sname$

$Cid \rightarrow Cname$

Updation Anomaly :-

Updation is required in all duplicate Copies, ∴ it will be time consuming operation

Topic : Schema refinement (Normalization)

* Normalization is a process of **decomposing (splitting)** a relational tables into smaller tables (sub-relations) such that it eliminates/reduces the data redundancy, and it can overcome undesirable characteristics like Insertion, Updation and Deletion Anomalies

Student_Enrollment_Info

$F = \begin{Bmatrix} Sid \rightarrow Sname \\ Cid \rightarrow Cname \end{Bmatrix}$

Sid	Sname	Cid	Cname
S_1	A	C_1	DM
S_1	A	C_2	DBMS
S_2	A	C_2	DBMS
S_3	B	C_2	DBMS
S_3	B	C_3	DM

Assume that the relation is decomposed into following three sub-relations

Student

Sid	Sname
S_1	A
S_2	A
S_3	B

$F_1 = \{Sid \rightarrow Sname\}$

Ck = Sid

Course

Cid	Cname
C_1	DM
C_2	DBMS
C_3	DM
C_4	AI

$F_2 = \{Cid \rightarrow Cname\}$

Ck = Cid

Enroll

Sid	Cid
S_1	C_1
S_1	C_2
S_2	C_2
S_3	C_2
S_3	C_3

$F_3 = \{ \quad \}$

Ck = (Sid, Cid)

After decomposition there is no redundancy

&

all the anomalies are overcome

Topic : Properties of decomposition

While decomposing a relation in to sub-relations following properties must be ensured

① Decomposition must be dependency preserving

i.e; All the functional dependencies present in the original relation must be preserved even after the decomposition into sub-relations.

② Decomposition must be Lossless Join decomposition

i.e, When we perform the natural join (⋈) of all the sub-relations, then we must get the Exact same tuples as of original relation.

i.e, After decomposition there should not be any loss of information

Neither in terms of FDs

Nor in terms of tuples

Properties of decomposition

1) Dependency preserving decomposition

- Let R be the relational schema with FD set F is decomposed into sub-relation. $R_1, R_2, R_3 \ldots R_m$ with FD sets $F_1, F_2, F_3, \ldots F_m$ respectively

i.e. $R(F)$ → $R_1(F_1)$, $R_2(F_2)$, $R_3(F_3) \ldots$ $R_m(F_m)$

In general, $F_1 \cup F_2 \cup F_3 \cup \ldots \cup F_m \subseteq F$

If, $F_1 \cup F_2 \cup F_3 \cup \ldots \cup F_m = F$, then dep. preserving

If, $F_1 \cup F_2 \cup F_3 \cup \ldots \cup F_m \subset F$, not dep. preserving

$\boxed{F_1 \cup F_2 \cup F_3 \cup \ldots \cup F_m \supset F}$ Not possible

2) Lossless Join decomposition

Let R be the relational schema with FD set F is decomposed into sub-relation. $R_1, R_2, R_3 \ldots R_m$ with FD sets $F_1, F_2, F_3, \ldots F_m$ respectively

i.e. $R(F)$ → $R_1(F_1)$, $R_2(F_2)$, $R_3(F_3) \ldots$ $R_m(F_m)$

In general $R_1 \bowtie R_2 \bowtie \ldots \bowtie R_m \supseteq R$

if, "$=$", then lossless Join decomposition

if, "$\supset$" then lossy join decomposition

"$\subset$" not possible

Dependency Preserving decomposition

Q :- Let $R(A, B, C, D)$ be the relational schema with FD set $F = \{A \rightarrow B, B \rightarrow C, C \rightarrow D, D \rightarrow A\}$

If relation R is decomposed into three sub-relations $R_1(A,B)$, $R_2(B,C)$, & $R_3(C,D)$, then identify whether the decomposition is dependency preserving decomposition or not ?

Q :- Let $R(A, B, C, D)$ be the relational schema with FD set $F = \{A \to B,\ B \to C,\ C \to D,\ D \to A\}$

$R(A, B, C, D)$

$R_1(AB)$ $\quad$ $R_2(BC)$ $\quad$ $R_3(C,D)$

$F_1 = \{A \to B,\ B \to A\}$ $\quad$ $F_2 = \{B \to C,\ C \to B\}$ $\quad$ $F_3 = \{C \to D,\ D \to C\}$

$F_1 \cup F_2 \cup F_3 = \{A \to B,\ B \to C,\ C \to D,\ B \to A,\ C \to B,\ D \to C\}$

Check the relationship b/w "$F_1 \cup F_2 \cup F_3$" and F

- $F_1 \cup F_2 \cup F_3 \subseteq F$ — eqⁿ ① → it will always hold true
- Check if $F \subseteq F_1 \cup F_2 \cup F_3$ or not

FDs of F

$A \to B$ directly present in $F_1 \cup F_2 \cup F_3$

$B \to C$ directly present in $F_1 \cup F_2 \cup F_3$

$C \to D$ directly present in $F_1 \cup F_2 \cup F_3$

$D \to A$, $(D)^+_{\text{w.r.t. } F_1 \cup F_2 \cup F_3} = \{D, C, B, A\}$. $A \in D^+$ $\therefore D \to A$ is preserved

Hence $F \subseteq F_1 \cup F_2 \cup F_3$ — eqⁿ ②

By eqⁿ 1 & eqⁿ 2

$$F_1 \cup F_2 \cup F_3 = F$$

$\therefore$ Decomposition is dependency Preserving

Q :- Let $R(A, B, C, D)$ be the relational schema with FD set $F = \{AB \rightarrow CD, D \rightarrow A\}$

If relation R is decomposed into two sub-relations $R_1(A, D)$ & $R_2(BCD)$, then identify whether the decomposition is dependency preserving decomposition or not ?

$R(ABCD) \Rightarrow F = \{AB \to CD,\ D \to A\}$

$R_1(AD)$ $\quad$ $R_2(BCD)$

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

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

$F_1 = \{D \to A\}$

$(B)^+ = \{\cancel{B}\}$

$(C)^+ = \{\cancel{C}\}$

$(D)^+ = \{\cancel{D}, \cancel{A}\}$

$(BC)^+ = \{\cancel{B}, \cancel{C}\}$

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

$(CD)^+ = \{\cancel{C}, \cancel{D}, \cancel{A}\}$

$F_2 = \{BD \to C\}$

$$F_1 \cup F_2 = \begin{Bmatrix} D \to A \\ BD \to C \end{Bmatrix}$$

- We know $F_1 \cup F_2 \subseteq F$ — ①
- Check if $F \subseteq F_1 \cup F_2$ or not?

$AB \to CD \Rightarrow (AB)^+_{\text{w.r.t. } F_1 \cup F_2} = \{A, B\}$

$CD \notin \{AB\}^+$

$\therefore AB \to CD$ is lost

$\therefore F \not\subseteq F_1 \cup F_2$ — eqn ②

By ① & ②

$F \subset F_1 \cup F_2$

Hence, Not dep. preserving decomposition

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

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

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

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

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.

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

Topic : Lossless Join decomposition

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

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

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

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

$R1 \bowtie R2 \bowtie .... \bowtie Rn \subset R$ (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 (×)

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'

ie 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×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}

$$R \bowtie S = \pi_{R.A, R.B, R.C, S.D, S.E} \; (R \times S) \quad \therefore \equiv R \times S$$

Project all attributes of $R \times S$

No Common attributes, $\therefore$ No Selection Cond^n.
Hence, all the tuples of $R \times S$ will be Selected

i.e. Select all tuples of $R \times S$ & Project all attributes of $R \times S$

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

H.W. Consider the following relational table.

R

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

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

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

2 mins Summary

Topic: Properties of decomposition

Topic: Dependency preserving decomposition

Topic: Lossless Join decomposition

Slide

THANK - YOU

Slide