CS & IT ENGINEERING

Database Management System

Set Operations

Lecture No. 6

By- Ravindrababu Ravula Sir

Recap of Previous Lecture

Topic

Queues

Topics to be Covered

after SQL

- **Topic** Integrity Constraints & ER Model (2 Marks)
- **Topic** Normalization (2-4 Marks)
- **Topic** Queries (Relational Algebra, SQL, Tuple Relational Calculus) (4 Marks)
- **Topic** File Organization & Indexing(2-4 Marks)
- **Topic** Transactions & Concurrency Control (2- 4 Marks)

Topics to be Covered

- **Topic** Queries
- **Topic** Relational Algebra
- **Topic** SOME,EVERY,ATLEAST/Atmost/exactly
- **Topic** MAXIMUM, MINIMUM

Topic: Set Operations

SET OPERATIONS:

- Union (U)
- Set difference (–)
- Intersection (∩)

Topic: Set Operations

SET OPERATIONS:

- Union (U)
- Set difference (–)
- Intersection (∩)

➢ To apply any of set operations on two relations R and S, both R and S must be union compatible.

Topic: Set Operations

Union Compatibility:

- R and S are union compatible iff
  (i) Arity of R = Arity of S

Topic: Set Operations

Union Compatibility:

- R and S are union compatible iff
  - (i) Arity of R = Arity of S
    and
  - (ii) Possible values domain of each attribute of R must be same as attributes of S respectively.

Topic: Set Operations

Example:-1

$\Pi_{SidSname}(....)$ $\Pi_{Sid}(....)$

Topic: Set Operations

Example:-1

$\Pi_{SidSname}(....)$ $\Pi_{Sid}(....)$

2 1

Both are not union compatible because arity is different .

Topic: Set Operations

Example:-1

$\Pi_{SidSname}(....)$ $\Pi_{Sid}(....)$

Both are not union compatible because arity is different .

Example:-2

$\Pi_{SidSname}(....)$ $\Pi_{Sidage}(....)$

char numeric

Topic: Set Operations

Example:-1

$\Pi_{SidSname}(....)$ $\Pi_{Sid}(....)$

Both are not union compatible because arity is different .

Example:-2

$\Pi_{SidSname}(....)$ $\Pi_{Sidage}(....)$

Arity is same, but domains are different.

Topic: Set Operations

Example:-1

$\Pi_{SidSname}(....)$ $\Pi_{Sid}(....)$

Both are not union compatible because arity is different .

Example:-2

$\Pi_{SidSname}(....)$ $\Pi_{Sidage}(....)$

Arity is same, but domains are different.

Example:-3

$\Pi_{SidSname}(....)$ $\Pi_{StudID, StudName}(....)$

2 2

Topic: Set Operations

Example:-1

$\Pi_{SidSname}(....)$ $\Pi_{Sid}(....)$

Both are not union compatible because arity is different .

Example:-2

$\Pi_{SidSname}(....)$ $\Pi_{Sidage}(....)$

Arity is same, but domains are different.

Example:-3

$\Pi_{SidSname}(....)$ $\Pi_{StudID,\ StudName}(....)$

Set Operations are allowed.

Topic: Set Operations

- For $R \cup S$, $R \cap S$, $R - S$ set operations.

Topic: Set Operations

- For $R \cup S$, $R \cap S$, $R - S$ set operations.
  (i) Schema for result is same as schema of R.

Topic: Set Operations

- For $R \cup S$, $R \cap S$, $R - S$ set operations.
  (i) Schema for result is same as schema of R.
  (ii) Resulted record set is distinct record set.

Topic: Set Operations

- For $R \cup S$, $R \cap S$, $R - S$ set operations.
  (i) Schema for result is same as schema of R.
  (ii) Resulted record set is distinct record set.
- These constraints are same in SQL

Topic: Set Operations

Example:-

R

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

S

D	E	F
2	4	6
2	4	6
4	5	6
3	5	9

What is $R \cup S$, $R \cap S$, $R - S$?

Topic: Set Operations

Example:-

R

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

S

D	E	F
2	4	6
2	4	6
4	5	6
3	5	9

What is $R \cup S$, $R \cap S$, $R - S$?

Sol: $R \cup S \equiv R$

Topic: Set Operations

Example:-

R

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

S

D	E	F
2	4	6
2	4	6
4	5	6
3	5	9

What is $R \cup S$, $R \cap S$, $R - S$?

Sol: $R \cup S \equiv R$

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

Either records from R or S

Topic: Set Operations

Example:-

P

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

S

D	E	F
2	4	6
2	4	6
4	5	6
3	5	9

What is $R \cup S$, $R \cap S$, $R - S$?

Sol: $R \cup S \equiv R$ $R \cap S \equiv R$

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

Either records from R or S

Topic: Set Operations

Example:-

P

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

S

D	E	F
2	4	6
2	4	6
4	5	6
3	5	9

What is $R \cup S$, $R \cap S$, $R - S$?

Sol:

$R \cup S \equiv R$

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

Either records from R or S

$R \cap S \equiv R$

A	B	C
2	4	6
4	5	6

Records from R and S

Topic: Set Operations

Example:-

P

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

S

D	E	F
2	4	6
2	4	6
4	5	6
3	5	9

What is $R \cup S$, $R \cap S$, $R - S$?

Sol:

$R \cup S \equiv R$

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

Either records from R or S

$R \cap S \equiv R$

R

A	B	C
2	4	6
4	5	6

Records from R and S

$R - S \equiv R$

Topic: Set Operations

Example:-

P

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

S

D	E	F
2	4	6
2	4	6
4	5	6
3	5	9

What is $R \cup S$, $R \cap S$, $R - S$?

Sol:

$R \cup S \equiv R$

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

Either records from R or S

$R \cap S \equiv R$

A	B	C
2	4	6
4	5	6

Records from R and S

$R - S \equiv R$

A	B	C
3	5	7

Records of R but not S [only records of R]

Topic: Set Operations

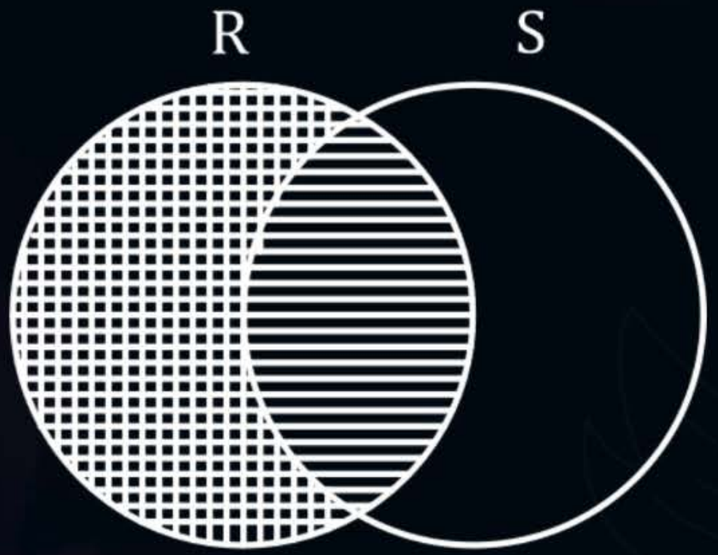
- Union ($\cup$), Set Difference($-$) are basic operators

Topic: Set Operations

- Union ($\cup$), Set Difference($-$) are basic operators
- $\cap$ is derived operator

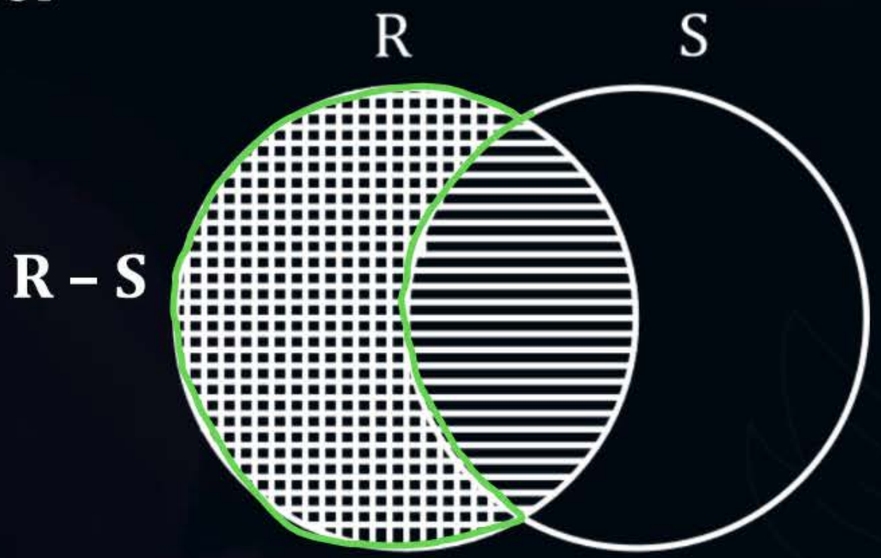
Topic: Set Operations

- Union ($\cup$), Set Difference($-$) are basic operators
- $\cap$ is derived operator



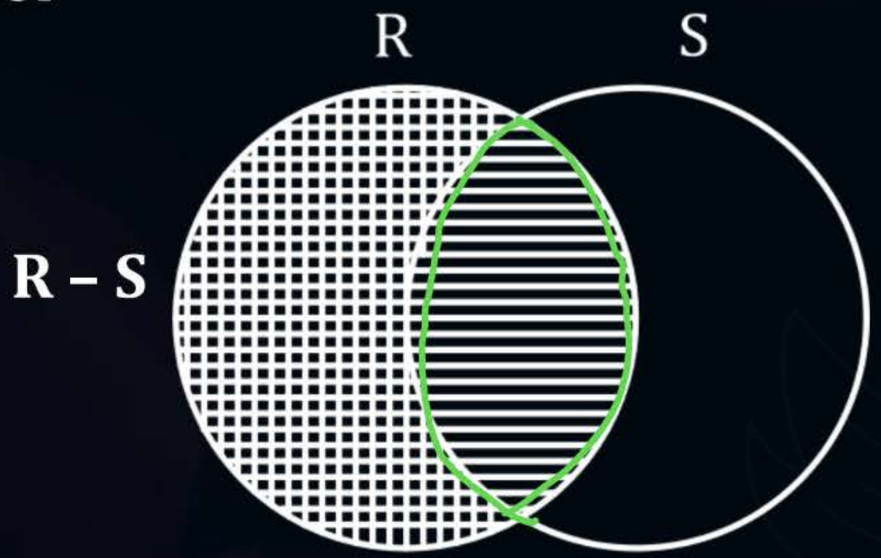
Topic: Set Operations

- Union ($\cup$), Set Difference($-$) are basic operators
- $\cap$ is derived operator



Topic: Set Operations

- Union ($\cup$), Set Difference($-$) are basic operators
- $\cap$ is derived operator



- $R \cap S = R - (R - S)$

Topic: Set Operations

For R(A,B) and S(A, B), Attributes names of R & S is same

Topic: Set Operations

For R(A,B) and S(A, B), Attributes names of R ∪ S is same

(i) $R \cap S = R - (R - S)$

Topic: Set Operations

For R(A,B) and S(A, B), Attributes names of $R \cup S$ is same

(i) $R \cap S = R - (R - S)$

(ii) $R \cap S = R \bowtie S$

$R.A = S.A$

$R.B = S.B$

Topic: Set Operations

For R(A,B) and S(A, B), Attributes names of $R \cup S$ is same

(i) $R \cap S = R - (R - S)$

(ii) $R \cap S = R \bowtie S$

R

A	B
4	6
2	5

S

A	B
4	6
2	8

$\Rightarrow$

$R \bowtie S$

A	B
4	6

Topic: Set Operations

For R(A,B) and S(A, B), Attributes names of $R \cup S$ is same

(i) $R \cap S = R - (R - S)$

(ii) $R \cap S = R \bowtie S$

R

A	B
4	6
2	5

S

A	B
4	6
2	8

$\Rightarrow$ $R \bowtie S$

A	B
4	6

(iii) $R \cup S = R \bowtie S$

Topic: Set Operations

For R(A,B) and S(A, B), Attributes names of $R \cup S$ is same

(i) $R \cap S = R - (R - S)$

(ii) $R \cap S = R \bowtie S$

R

A	B
4	6
2	5

S

A	B
4	6
2	8

$\Rightarrow$ $R \bowtie S$

A	B
4	6

(iii) $R \cup S = R \bowtie S$

A	B
4	6
2	5
2	8

Topic: Set Operations

#Q. R(AB) S(CD)

Comparison btn $R \bowtie S$, $R \cup S$, $R \cap S$

R

A	B
4	6
2	5

S

C	D
4	6
2	8

Topic: Set Operations

#Q. R(AB) S(CD)

Comparison btn $R \bowtie S$, $R \cup S$, $R \cap S$

R

A	B
4	6
2	5

S

C	D
4	6
2	8

A $R \cap S = R - (R - S)$

Topic: Set Operations

#Q. R(AB) S(CD)

Comparison btn $R \bowtie S$, $R \cup S$, $R \cap S$

R

A	B
4	6
2	5

S

C	D
4	6
2	8

A $R \cap S = R - (R - S)$

B $R \bowtie S \neq R \cap S$

No common Attribute

Topic: Set Operations

#Q. R(AB) S(CD)

Comparison btn $R \bowtie S$, $R \cup S$, $R \cap S$

R

A	B
4	6
2	5

S

C	D
4	6
2	8

A $R \cap S = R - (R - S)$

B $R \bowtie S \neq R \cap S$

$R \bowtie S = R \times S$

No common Attribute

Topic: Set Operations

#Q. R(AB) S(CD)

Comparison btn $R \bowtie S$, $R \cup S$, $R \cap S$

R

A	B
4	6
2	5

S

C	D
4	6
2	8

A $R \cap S = R - (R - S)$

B

$R \bowtie S \neq R \cap S$

$R \bowtie S = R \times S$

No common Attribute

Topic: Set Operations

#Q. R(AB) S(CD)

Comparison btn $R \bowtie S$, $R \cup S$, $R \cap S$

R

A	B
4	6
2	5

S

C	D
4	6
2	8

A B

A $R \cap S = R - (R - S)$

B $R \bowtie S \neq R \cap S$

$R \bowtie S = R \times S$

No common Attribute

C $R \cap S = R \bowtie \rho_{C \to A,\ D \to B}(S)$

Topic: Set Operations

#Q. R(AB) S(CD)

Comparison btn $R \bowtie S$, $R \cup S$, $R \cap S$

R

A	B
4	6
2	5

S

C	D
4	6
2	8

A $R \cap S = R - (R - S)$

B $R \bowtie S \neq R \cap S$

$R \bowtie S = R \times S$

No common Attribute

C $R \cap S = R \bowtie \rho_{C \to A,\ D \to B}(S)$

D $R \cup S \neq R \bowtie S$, No common Attributes

Topic: Set Operations

#Q. R(AB) S(CD)

Comparison btn $R \bowtie S$, $R \cup S$, $R \cap S$

R

A	B
4	6
2	5

S

C	D
4	6
2	8

A $R \cap S = R - (R - S)$

B $R \bowtie S \neq R \cap S$

$R \bowtie S = R \times S$

No common Attribute

C $R \cap S = R \bowtie \rho_{C \to A,\ D \to B}(S)$

D $R \cup S \neq R \bowtie S$, No common Attributes

$\Downarrow$

R×S

Topic: Set Operations

#Q. R(AB) S(CD)

Comparison btn $R \bowtie S$, $R \cup S$, $R \cap S$

R

A	B
4	6
2	5

S

C	D
4	6
2	8

A B

A $R \cap S = R - (R - S)$

B $R \bowtie S \neq R \cap S$
$R \bowtie S = R \times S$
No common Attribute

C $R \cap S = R \bowtie \rho_{C \to A,\ D \to B}(S)$

D $R \cup S \neq R \bowtie S$, No common Attributes

⇓

$R \times S$

E $R \cup S = R \bowtie \rho_{C \to A, D \to B}(S)$

Inspiring Stories : Noor Inayat Khan

Background: **Indian-origin woman in UK during WWII.**

Education: **Trained as a musician and writer.**

Achievements: **Became a secret agent, caught and killed by Nazis.**

Impact: **Symbol of courage and sacrifice.**

Topic: Set Operations

$\Pi_{AB}(R) / \Pi_{B}(S)$

Topic: Set Operations

$$\Pi_{AB}(R) / \Pi_{B}(S) = \Pi_{A}(R) - \Pi_{A}\left(\Pi_{A}(R) \times \Pi_{B}(S) - \Pi_{AB}(R)\right)$$

Topic: Set Operations

$$\Pi_{AB}(R) / \Pi_{B}(S) = \Pi_{A}(R) - \Pi_{A}\left(\Pi_{A}(R) \times \Pi_{B}(S) - \Pi_{AB}(R)\right)$$

AB AB

$\cup$ compatible

Topic: Set Operations

$$\Pi_{AB}(R) / \Pi_{B}(S) = \Pi_{A}(R) - \Pi_{A}\left(\Pi_{A}(R) \times \Pi_{B}(S) - \Pi_{AB}(R)\right)$$

Union compatible

$\cup$ compatible

Topic: Set Operations

(i) Retrieve Sid's enrolled for some course taught by KORTH or some course taught by Navathe.

Stud

Sid	Sname	Age
S1	A	-
S2	B	-
S3	C	-
S4	D	-

Course

Cid	Cname	Intr
C1	DB	Korth
C2	DB	Korth
C3	DB	Navathe
C4	DB	Navathe
D5	OS	Galwin

Enroll

Sid	Cid	Fee
S1	C1	-
S1	C2	-
S1	C3	-
S1	C4	-
S2	C1	-
S2	C2	-
S1	C5	-
S3	C3	-
S4	C5	-

S4 C5

Topic: Set Operations

(i) Retrieve Sid's enrolled for some course taught by KORTH or some course taught by Navathe.

Query 1:

Stud

Sid	Sname	Age
S1	A	-
S2	B	-
S3	C	-
S4	D	-

Course

Cid	Cname	Intr
C1	DB	Korth
C2	DB	Korth
C3	DB	Navathe
C4	DB	Navathe
D5	OS	Galwin

Enroll

	Sid	Cid	Fee
✓	S1	C1	-
✓	S1	C2	-
✓	S1	C3	-
✓	S1	C4	-
✓	S2	C1	-
✓	S2	C2	-
✓	S1	C5	-
✓	S3	C3	-
×	S4	C5	-

Topic: Set Operations

(i) Retrieve Sid's enrolled for some course taught by KORTH or some course taught by Navathe.

Query 1: $\sigma_{Inst=KORTH \vee Inst=Navathe}(Course)$

Stud

Sid	Sname	Age
S1	A	-
S2	B	-
S3	C	-
S4	D	-

Course

Cid	Cname	Intr
C1	DB	Korth
C2	DB	Korth
C3	DB	Navathe
C4	DB	Navathe
D5	OS	Galwin

Enroll

	Sid	Cid	Fee
✓	S1	C1	-
✓	S1	C2	-
✓	S1	C3	-
✓	S1	C4	-
✓	S2	C1	-
✓	S2	C2	-
✓	S1	C5	-
✓	S3	C3	-
×	S4	C5	-

Topic: Set Operations

(i) Retrieve Sid's enrolled for some course taught by KORTH or some course taught by Navathe.

Query 1: Π_{Sid} (Enroll $\bowtie \sigma_{Inst=KORTH \vee Inst=Navathe}$(Course))

Stud

Sid	Sname	Age
S1	A	-
S2	B	-
S3	C	-
S4	D	-

Course

Cid	Cname	Intr
C1	DB	Korth
C2	DB	Korth
C3	DB	Navathe
C4	DB	Navathe
D5	OS	Galwin

Enroll

	Sid	Cid	Fee
✓	S1	C1	-
✓	S1	C2	-
✓	S1	C3	-
✓	S1	C4	-
✓	S2	C1	-
✓	S2	C2	-
✓	S1	C5	-
✓	S3	C3	-
×	S4	C5	-

Topic: Set Operations

(i) Retrieve Sid's enrolled for some course taught by KORTH or some course taught by Navathe.

Query 1: Π_{Sid} (Enroll $\bowtie \sigma_{Inst=KORTH \vee Inst=Navathe}$(Course))

Query 2: $\sigma_{Inst=KORTH}$ (Course) $\sigma_{Inst=Navathe}$ (Course)

Stud

Sid	Sname	Age
S1	A	-
S2	B	-
S3	C	-
S4	D	-

Course

Cid	Cname	Intr
C1	DB	Korth
C2	DB	Korth
C3	DB	Navathe
C4	DB	Navathe
D5	OS	Galwin

Enroll

	Sid	Cid	Fee
✓	S1	C1	-
✓	S1	C2	-
✓	S1	C3	-
✓	S1	C4	-
✓	S2	C1	-
✓	S2	C2	-
✓	S1	C5	-
✓	S3	C3	-
×	S4	C5	-

Topic: Set Operations

(i) Retrieve Sid's enrolled for some course taught by KORTH or some course taught by Navathe.

Query 1: Π_{Sid} (Enroll $\bowtie \sigma_{Inst=KORTH \vee Inst=Navathe}$(Course))

Query 2: Enroll $\bowtie \sigma_{Inst=KORTH}$ (Course) Enroll $\bowtie \sigma_{Inst=Navathe}$ (Course)

Stud

Sid	Sname	Age
S1	A	-
S2	B	-
S3	C	-
S4	D	-

Course

Cid	Cname	Intr
C1	DB	Korth
C2	DB	Korth
C3	DB	Navathe
C4	DB	Navathe
D5	OS	Galwin

Enroll

	Sid	Cid	Fee
✓	S1	C1	-
✓	S1	C2	-
✓	S1	C3	-
✓	S1	C4	-
✓	S2	C1	-
✓	S2	C2	-
✓	S1	C5	-
✓	S3	C3	-
×	S4	C5	-

Topic: Set Operations

(i) Retrieve Sid's enrolled for some course taught by KORTH or some course taught by Navathe.

Query 1: Π_{Sid} (Enroll $\bowtie \sigma_{Inst=KORTH \vee Inst=Navathe}$(Course))

Query 2: Π_{Sid} (Enroll $\bowtie \sigma_{Inst=KORTH}$ (Course)) Π_{Sid} (Enroll $\bowtie \sigma_{Inst=Navathe}$ (Course))

Stud

Sid	Sname	Age
S1	A	-
S2	B	-
S3	C	-
S4	D	-

Course

Cid	Cname	Intr
C1	DB	Korth
C2	DB	Korth
C3	DB	Navathe
C4	DB	Navathe
D5	OS	Galwin

Enroll

	Sid	Cid	Fee
✓	S1	C1	-
✓	S1	C2	-
✓	S1	C3	-
✓	S1	C4	-
✓	S2	C1	-
✓	S2	C2	-
✓	S1	C5	-
✓	S3	C3	-
×	S4	C5	-

Topic: Set Operations

(i) Retrieve Sid's enrolled for some course taught by KORTH or some course taught by Navathe.

Query 1: Π_{Sid} (Enroll $\bowtie \sigma_{Inst=KORTH \vee Inst=Navathe}$(Course))

Query 2: Π_{Sid} (Enroll $\bowtie \sigma_{Inst=KORTH}$ (Course)) $\cup$ Π_{Sid} (Enroll $\bowtie \sigma_{Inst=Navathe}$ (Course))

Stud

Sid	Sname	Age
S1	A	-
S2	B	-
S3	C	-
S4	D	-

Course

Cid	Cname	Intr
C1	DB	Korth
C2	DB	Korth
C3	DB	Navathe
C4	DB	Navathe
C5	OS	Galwin

Enroll

	Sid	Cid	Fee
✓	S1	C1	-
✓	S1	C2	-
✓	S1	C3	-
✓	S1	C4	-
✓	S2	C1	-
✓	S2	C2	-
✗	S1	C5	-
✓	S3	C3	-
✗	S4	C5	-

Topic: Set Operations

(i) Retrieve Sid's enrolled for some course taught by KORTH or some course taught by Navathe.

Query 1: Π_{Sid} (Enroll $\bowtie \sigma_{Inst=KORTH \vee Inst=Navathe}$(Course))

Query 2: Π_{Sid} (Enroll $\bowtie \sigma_{Inst=KORTH}$ (Course)) $\cup$ Π_{Sid} (Enroll $\bowtie \sigma_{Inst=Navathe}$ (Course))

or

Stud

Sid	Sname	Age
S1	A	-
S2	B	-
S3	C	-
S4	D	-

Course

Cid	Cname	Intr
C1	DB	Korth
C2	DB	Korth
C3	DB	Navathe
C4	DB	Navathe
D5	OS	Galwin

Enroll

	Sid	Cid	Fee
✓	S1	C1	-
✓	S1	C2	-
✓	S1	C3	-
✓	S1	C4	-
✓	S2	C1	-
✓	S2	C2	-
✓	S1	C5	-
✓	S3	C3	-
×	S4	C5	-

Topic: Set Operations

(ii) Retrieve Sids enrolled for some course taught by KORTH and some course by Navathe.

Query 1: $\Pi_{Sid}(Enroll \bowtie \sigma_{Inst=KORTH \wedge Inst=Navathe}(Course))$

Topic: Set Operations

(ii) Retrieve Sids enrolled for some course taught by KORTH and some course by Navathe.

Query 1: $\Pi_{Sid}(Enroll \bowtie \sigma_{Inst=KORTH \wedge Inst=Navathe}(Course))$

C1 Korth (×)
C2 Korth (×)
C3 Navathe (×)
C3 Navathe (×)

Topic: Set Operations

(ii) Retrieve Sids enrolled for some course taught by KORTH and some course by Navathe.

Query 1: Π_{Sid}(Enroll $\bowtie$ $\sigma_{Inst=KORTH \wedge Inst=Navathe}$(Course))

Empty record set

C1 Korth (×)
C2 Korth (×)
C3 Navathe(×)
C3 Navathe(×)

Topic: Set Operations

(ii) Retrieve Sids enrolled for some course taught by KORTH and some course by Navathe.

Query 1: Π_{Sid}(Enroll $\bowtie$ $\sigma_{Inst=KORTH \wedge Inst=Navathe}$(Course))

Empty record set ✓

Empty record set

C1 Korth (×)
C2 Korth (×)
C3 Navathe(×)
C3 Navathe(×)

Topic: Set Operations

(ii) Retrieve Sids enrolled for some course taught by KORTH and some course by Navathe.

Query 1: Π_{Sid}(Enroll $\bowtie$ $\sigma_{Inst=KORTH \wedge Inst=Navathe}$(Course))

Empty record set

Empty record set

C1 Korth (×)
C2 Korth (×)
C3 Navathe(×)
C3 Navathe(×)

- Result of $\bowtie$ will be empty

$\therefore$ The query is wrong.

Topic: Set Operations

(ii) Retrieve Sids enrolled for some course taught by KORTH and some course by Navathe.

Query 1: $\Pi_{Sid}(Enroll \bowtie \sigma_{Inst=KORTH \wedge Inst=Navathe}(Course))$

Query-2: $\Pi_{Sid}(Enroll \bowtie \sigma_{Inst=KORTH}(Course)) \cap \Pi_{Sid}(Enroll \bowtie \sigma_{Inst=Navathe}(Course))$

Topic: Set Operations

(ii) Retrieve Sids enrolled for some course taught by KORTH and some course by Navathe.

Query 1: $\Pi_{Sid}(Enroll \bowtie \sigma_{Inst=KORTH \wedge Inst=Navathe}(Course))$ ✗

Query-2: $\Pi_{Sid}(Enroll \bowtie \sigma_{Inst=KORTH}(Course)) \cap \Pi_{Sid}(Enroll \bowtie \sigma_{Inst=Navathe}(Course))$

S1
S2 ✓

S1
S3 ✓

Topic: Set Operations

(ii) Retrieve Sids enrolled for some course taught by KORTH and some course by Navathe.

Query 1: $\Pi_{Sid}(Enroll \bowtie \sigma_{Inst=KORTH \wedge Inst=Navathe}(Course))$ ✗

Query-2: $\Pi_{Sid}(Enroll \bowtie \sigma_{Inst=KORTH}(Course)) \cap \Pi_{Sid}(Enroll \bowtie \sigma_{Inst=Navathe}(Course))$

S1
S2

S1
S3

Sid
S1

Topic: Set Operations

(ii) Retrieve Sids enrolled for some course taught by KORTH and some course by Navathe.

Query 1: $\Pi_{Sid}(Enroll \bowtie \sigma_{Inst=KORTH \wedge Inst=Navathe}(Course))$ ✗

Query-2: $\Pi_{Sid}(Enroll \bowtie \sigma_{Inst=KORTH}(Course)) \cap \Pi_{Sid}(Enroll \bowtie \sigma_{Inst=Navathe}(Course))$

S1
S2

S1
S3

Sid
S1

This is the correct Query ✓

$\therefore$ $\cup$ is used for OR $\cap$ is used for AND

Topic: Set Operations

#Q. Enroll (Sid Cid) Course (Cid Instr)

(ii) Retrieve Sids enrolled only course taught by KORTH.

Enroll

Sid	Cid
S1	C1
S1	C2
S1	C4
S2	C2
S3	C1
S3	C2
S4	C5

Course

Cid	Intr
C1	Korth
C2	Korth
C3	Navathe
C4	Navathe
C5	Uuman

Topic: Set Operations

#Q. Enroll (Sid Cid) Course (Cid Instr)

(ii) Retrieve Sids enrolled only (set difference) course taught by KORTH.

Enroll

Sid	Cid
S1	C1
S1	C2
S1	C4
S2	C2
S3	C1
S3	C2
S4	C5

Course

Cid	Intr
C1	Korth
C2	Korth
C3	Navathe
C4	Navathe
C5	Uuman

Topic: Set Operations

#Q. Enroll (Sid Cid) Course (Cid Instr)

(ii) Retrieve Sids enrolled only (set difference) course taught by KORTH.

Sol: S1 (✗) S2 (✓) S3 (✓) S4 (✗)

Enroll

Sid	Cid
S1	C1
S1	C2
S1	C4
S2	C2
S3	C1
S3	C2
S4	C5

Course

Cid	Intr
C1	Korth
C2	Korth
C3	Navathe
C4	Navathe
C5	Uuman

Topic: Set Operations

#Q. Enroll (Sid Cid) Course (Cid Instr)

(ii) Retrieve Sids enrolled only (set difference) course taught by KORTH.

Sol: S1 (✗) S2 (✓) S3 (✓) S4 (✗)

- {Sid's enrolled} – {Sid's enrolled}
  some course — some non KORTH course

Enroll

Sid	Cid
S1	C1
S1	C2
S1	C4
S2	C2
S3	C1
S3	C2
S4	C5

Course

Cid	Intr
C1	Korth
C2	Korth
C3	Navathe
C4	Navathe
C5	Uuman

Topic: Set Operations

#Q. Enroll (Sid Cid) Course (Cid Instr)

(ii) Retrieve Sids enrolled only (set difference) course taught by KORTH.

Sol: S1 (✗) S2 (✓) S3 (✓) S4 (✗)

- {Sid's enrolled} – {Sid's enrolled}
  some course some non KORTH course

$\Rightarrow \quad \Pi_{Sid}(Enroll) - \Pi_{Sid}(Enroll \bowtie \sigma_{Inst \neq KORTH}(Course))$

S1, S2, S3, S4 – S1, S4 = S2, S3

Enroll

Sid	Cid
S1	C1
S1	C2
S1	C4
S2	C2
S3	C1
S3	C2
S4	C5

Course

Cid	Intr
C1	Korth
C2	Korth
C3	Navathe
C4	Navathe
C5	Uuman

Topic: Set Operations

#Q. Enroll (Sid Cid) Course (Cid Instr)

(ii) Retrieve Sids enrolled only (set difference) course taught by KORTH.

Sol: S1 (✗) S2 (✓) S3 (✓) S4 (✗)

- {Sid's enrolled} – {Sid's enrolled}
  some course some non KORTH course

$\Rightarrow \quad \Pi_{Sid}(Enroll) - \Pi_{Sid}(Enroll \bowtie \sigma_{Inst \neq KORTH}(Course))$

(ii) Some course taught by KORTH

Enroll

Sid	Cid
S1	C1
S1	C2
S1	C4
S2	C2
S3	C1
S3	C2
S4	C5

Course

Cid	Intr
C1	Korth
C2	Korth
C3	Navathe
C4	Navathe
C5	Uuman

Topic: Set Operations

#Q. Enroll (Sid Cid) Course (Cid Instr)

(ii) Retrieve Sids enrolled only (set difference) course taught by KORTH.

Sol: S1 (✗) S2 (✓) S3 (✓) S4 (✗)

- {Sid's enrolled} – {Sid's enrolled}
  some course some non KORTH course

$\Rightarrow \quad \Pi_{Sid}(Enroll) - \Pi_{Sid}(Enroll \bowtie \sigma_{Inst \neq KORTH}(Course))$

(ii) Some (✓) (⋈) course taught by KORTH

Enroll

Sid	Cid
S1	C1
S1	C2
S1	C4
S2	C2
S3	C1
S3	C2
S4	C5

Course

Cid	Intr
C1	Korth
C2	Korth
C3	Navathe
C4	Navathe
C5	Uuman

Topic: Set Operations

#Q. Enroll (Sid Cid) Course (Cid Instr)

(ii) Retrieve Sids enrolled only (set difference) course taught by KORTH.

Sol: S1 (✗) S2 (✓) S3 (✓) S4 (✗)

- {Sid's enrolled} – {Sid's enrolled}
  some course some non KORTH course

$\Rightarrow \quad \Pi_{Sid}(Enroll) - \Pi_{Sid}(Enroll \bowtie \sigma_{Inst \neq KORTH}(Course))$

Enroll

Sid	Cid
S1	C1
S1	C2
S1	C4
S2	C2
S3	C1
S3	C2
S4	C5

Course

Cid	Intr
C1	Korth
C2	Korth
C3	Navathe
C4	Navathe
C5	Uuman

(ii) Some ($\bowtie$) course taught by KORTH

$\Pi_{Sid}(Enroll) \bowtie \sigma_{Inst=KORTH}(Course)$

Topic: Set Operations

Query Optimization: (Asked for 4, 5 times in GATE)

- To optimize the time taken or cost occurrence of a query

Topic: Set Operations

Query Optimization: (Asked for 4, 5 times in GATE)

- To optimize the time taken or cost occurrence of a query

Example: On two Relations R(A,B) and S(B,C)

(i) $\sigma_{R.A > 10 \wedge R.B = S.B \wedge S.C > 5}(R \times S)$

Topic: Set Operations

Query Optimization: (Asked for 4, 5 times in GATE)

- To optimize the time taken or cost occurrence of a query

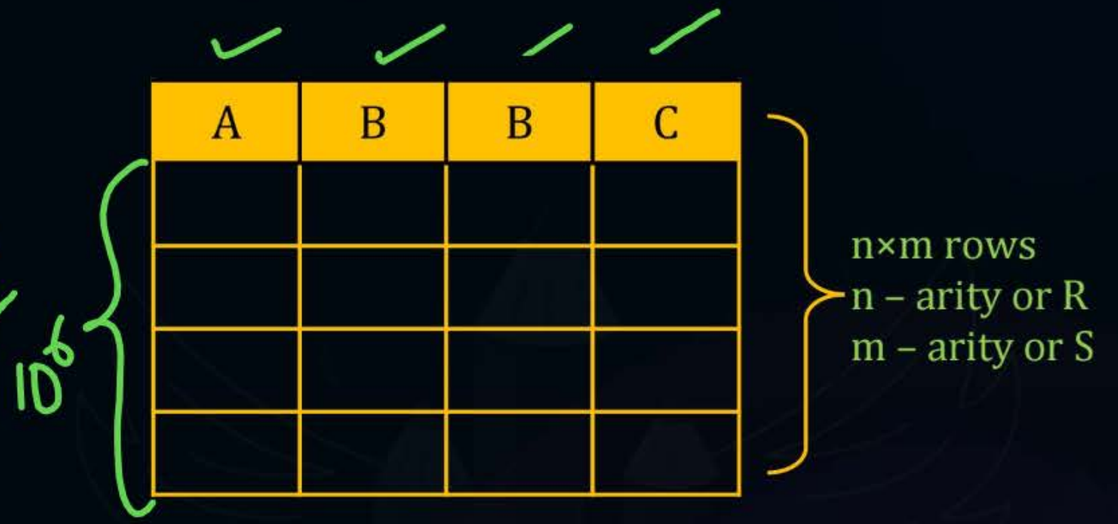
Example: On two Relations R(A,B) and S(B,C)

(i) $\sigma_{R.A > 10 \wedge R.B = S.B \wedge S.C > 5}(R \times S)$

1 2 3

$1000 \times 1000 = 10^6$ rows

3×10^6



Topic: Set Operations

Query Optimization: (Asked for 4, 5 times in GATE)

- To optimize the time taken or cost occurrence of a query

Example: On two Relations R(A,B) and S(B,C)

(i) $\sigma_{R.A > 10 \wedge R.B = S.B \wedge S.C > 5}(R \times S)$

R.A >10 $\Rightarrow$ can be before cross product

R.B = S.B $\Rightarrow$ should be done after cross product

S.C > 5 $\Rightarrow$ can be before cross product

A	B	B	C

n×m rows
n – arity or R
m – arity or S

Topic: Set Operations

Query Optimization: (Asked for 4, 5 times in GATE)

- To optimize the time taken or cost occurrence of a query

Example: On two Relations R(A,B) and S(B,C)

(i) $\sigma_{R.A > 10 \wedge R.B = S.B \wedge S.C > 5}(R \times S)$

R.A >10 $\Rightarrow$ can be before cross product

R.B = S.B $\Rightarrow$ should be done after cross product

S.C > 5 $\Rightarrow$ can be before cross product

A	B	B	C

n×m rows
n – arity or R
m – arity or S

$\sigma_{R.A>10 \wedge R.B=S.B \wedge S.C>5}(R \times S) \equiv \sigma_{R.B=S.B}(\sigma_{A>10}(R) \times \sigma_{C>5}(S))$

1000C 1000C

10 X 10 = 100C

2100C ✓

Topic: Set Operations

(ii) $\sigma_{R.A>10}(R \bowtie S)$

$1000 \times 1000 = 10^6$ rows

$= 10^6$ Comparisons

Topic: Set Operations

10 1000 = 10000C

(ii) $\sigma_{R.A>10}(R \bowtie S) \equiv \sigma_{A>10}(R) \bowtie S$

1000C

11000 C

Topic: Set Operations

(ii) $\sigma_{R.A>10}(R \bowtie S) \equiv \sigma_{A>10}(R) \bowtie S$

Join cost reduce = optimal

Topic: Set Operations

(ii) $\sigma_{R.A>10}(R \bowtie S) \equiv \sigma_{A>10}(R) \bowtie S$

Join cost reduce = optimal

(iii) $\sigma_{R.A>10 \wedge S.C>5}(R \bowtie S)$

Topic: Set Operations

(ii) $\sigma_{R.A>10}(R \bowtie S) \equiv \sigma_{A>10}(R) \bowtie S$

Join cost reduce = optimal

(iii) $\sigma_{R.A>10 \wedge S.C>5}(R \bowtie S)$

Can be added before join ✓

Topic: Set Operations

(ii) $\sigma_{R.A>10}(R \bowtie S) \equiv \sigma_{A>10}(R) \bowtie S$

Join cost reduce = optimal

(iii) $\sigma_{R.A>10 \wedge S.C>5}(R \bowtie S)$

Can be added
before join

$\equiv \sigma_{A>10}(R) \bowtie \sigma_{C>5}(S)$

Topic: Set Operations

(ii) $\sigma_{R.A>10}(R \bowtie S) \equiv \sigma_{A>10}(R) \bowtie S$

Join cost reduce = optimal

(iii) $\sigma_{R.A>10 \wedge S.C>5}(R \bowtie S)$

Can be added before join

$\equiv \sigma_{A>10}(R) \bowtie \sigma_{C>5}(S) \rightarrow$ Optimal, No. of Computations decreases

Inspiring Stories : A. Krishnamurthy

Background: **Techie from Chennai.**

Education: **Studied at Madras University.**

Achievements: **Quit job to clean polluted lakes, founded the Environmentalist Foundation of India.**

Impact: **Restored dozens of lakes in India.**

Topic: Set Operations

(iv) Enroll(Sid,Cid) → n Course(Cid,Instr) → m

$\sigma_{Instr=KORTH}(Enroll \bowtie Course)$

Topic: Set Operations

(iv) Enroll(Sid,Cid) $\rightarrow$ n Course(Cid,Instr) $\rightarrow$ m

$\sigma_{Instr=KORTH}$(Enroll$\bowtie$Course)

Sol: Let n =10000, m=100

Topic: Set Operations

(iv) Enroll(Sid,Cid) → n Course(Cid,Instr) → m

$\sigma_{Instr=KORTH}(Enroll \bowtie Course)$

Sol: Let n =10000, m=100

E.Cid = C.Cid

$\sigma_{Instr=KORTH}(Enroll \bowtie Course)$

i.e For ⋈ step1: Enroll.Cid = Course.Cid ⇒ 10^6 comparisons

Topic: Set Operations

(iv) Enroll(Sid,Cid) → n Course(Cid,Instr) → m

$\sigma_{Instr=KORTH}(Enroll \bowtie Course)$

Sol: Let n =10000, m=100

$\sigma_{Instr=KORTH}(\overbrace{Enroll \bowtie Course}^{1000000})$

1000000

i.e For ⋈ step1: Enroll.Cid = Course.Cid ⇒ 10^6 comparisons

step 2: Instr=KORTH 10^6 comparisons

Topic: Set Operations

(iv) Enroll(Sid,Cid) → n Course(Cid,Instr) → m

$\sigma_{Instr=KORTH}(Enroll \bowtie Course)$

Sol: Let n =10000, m=100

$$\underbrace{\sigma_{Instr=KORTH}(\overbrace{Enroll \bowtie Course}^{1000000})}_{1000000}$$

i.e For ⋈ step1: Enroll.Cid = Course.Cid ⇒ 10^6 comparisons

step 2: Instr=KORTH 10^6 comparisons

Total 2×10^6 comparisons

Topic: Set Operations

(iv) Enroll(Sid,Cid) → n Course(Cid,Instr) → m

$\sigma_{Instr=KORTH}(Enroll \bowtie Course)$

Sol: Let n =10000, m=100

1000000

$\sigma_{Instr=KORTH}(Enroll \bowtie Course)$

1000000

i.e For ⋈ step1: Enroll.Cid = Course.Cid ⇒ 10^6 comparisons

step 2: Instr=KORTH 10^6 comparisons

Total 2×10^6 comparisons

Now lets optimize

Enroll ⋈ $\sigma_{Instr=KORTH}$ (Course)

100 6m

10 X 1000

Topic: Set Operations

(iv) Enroll(Sid,Cid) → n Course(Cid,Instr) → m

$\sigma_{Instr=KORTH}(Enroll \bowtie Course)$

Sol: Let n =10000, m=100

$$\underbrace{\sigma_{Instr=KORTH}(\overbrace{Enroll \bowtie Course}^{1000000})}_{1000000}$$

i.e For ⋈ step1: Enroll.Cid = Course.Cid ⇒ 10^6 comparisons

step 2: Instr=KORTH 10^6 comparisons

Total 2×10^6 comparisons

Now lets optimize

Enroll ⋈ $\sigma_{Instr=KORTH}$ (Course) ⇒ 10^6Computations &result is lets say 5 tuples.

Topic: Set Operations

(iv) Enroll(Sid,Cid) → n Course(Cid,Instr) → m

$\sigma_{Instr=KORTH}$(Enroll⋈Course) n =10000, m=100

SolExt:

$\equiv \sigma_{E.cid=C.cid}$ (Enroll × $\sigma_{Instr=KORTH}$(Course))

100

100 comparisons

Topic: Set Operations

(iv) Enroll(Sid,Cid) → n Course(Cid,Instr) → m

$\sigma_{Instr=KORTH}$(Enroll⋈Course) n =10000, m=100

SolExt:

$\equiv \sigma_{E.cid=C.cid}$ (Enroll × $\sigma_{Instr=KORTH}$(Course))

10000

5 tuples

100 comparisons

Topic: Set Operations

(iv) Enroll(Sid,Cid) → n Course(Cid,Instr) → m

$\sigma_{Instr=KORTH}(Enroll \bowtie Course)$ n =10000, m=100

SolExt:

10000 5 tuples

$$\equiv \sigma_{E.cid=C.cid}\ (Enroll \times \sigma_{Instr=KORTH}(Course))$$

100 comparisons

Topic: Set Operations

(iv) Enroll(Sid,Cid) → n Course(Cid,Instr) → m

$\sigma_{Instr=KORTH}(Enroll \bowtie Course)$ n =10000, m=100

SolExt:

$$\equiv \sigma_{E.cid=C.cid}(Enroll \times \sigma_{Instr=KORTH}(Course))$$

10000

5 tuples

100 comparisons

50000 comp.

Topic: Set Operations

(iv) Enroll(Sid,Cid) → n Course(Cid,Instr) → m

$\sigma_{Instr=KORTH}(Enroll \bowtie Course)$ n =10000, m=100

SolExt:

10000 5 tuples

$\equiv \sigma_{E.cid=C.cid}(Enroll \times \sigma_{Instr=KORTH}(Course))$

100 comparisons

50000 comp.

Total = 50100 which is way less than 2000000.

Topic: Set Operations

(v) Emp(eid,dno,sal) $\Pi_{eid}(Emp \bowtie_{(sal>s \wedge D=5} \rho_{I,D,S}(Emp))$

Topic: Set Operations

(v) Emp(eid,dno,sal) $\Pi_{eid}(Emp \bowtie_{(sal>s \wedge D=5} \rho_{I,D,S}(Emp))$

Sol: Assume 100 emp && 10 from dept 5

$$\Pi_{eid}(Emp \bowtie_{(sal>s \wedge D=5} \rho_{I,D,S}(Emp))$$

Topic: Set Operations

(v) Emp(eid,dno,sal) $\Pi_{eid}(Emp \bowtie_{(sal>s \wedge D=5} \rho_{I,D,S}(Emp))$

Sol: Assume 100 emp && 10 from dept 5

$$\Pi_{eid}(Emp \bowtie_{(sal>s \wedge D=5} \rho_{I,D,S}(Emp))$$

100 tuples

100 tuples

Topic: Set Operations

(v) Emp(eid,dno,sal) $\Pi_{eid}(Emp \bowtie_{(sal>s \wedge D=5} \rho_{I,D,S}(Emp))$

Sol: Assume 100 emp && 10 from dept 5

$$\Pi_{eid}(Emp \bowtie_{(sal>s \wedge D=5} \rho_{I,D,S}(Emp))$$

100 tuples 100 tuples

10000 tuples for crossproduct

Topic: Set Operations

(v) Emp(eid,dno,sal) $\Pi_{eid}(Emp \bowtie_{(sal>s \wedge D=5} \rho_{I,D,S}(Emp))$

Sol: Assume 100 emp && 10 from dept 5

$$\Pi_{eid}(Emp \bowtie_{(sal>s \wedge D=5} \rho_{I,D,S}(Emp))$$

100 tuples 100 tuples

10000 tuples for crossproduct

Cost of Query: 20000 comparisons 10000 for Cproduct and another for condition

Topic: Set Operations

(v) Emp(eid,dno,sal) $\Pi_{eid}(Emp \bowtie_{(sal>s \wedge D=5} \rho_{I,D,S}(Emp))$ 100 emp && 10 from dept 5

Sol Ext: $\Pi_{eid}(Emp \bowtie_{sal>S} (\sigma_{D=5} (\rho_{I,D,S}(Emp))))$

Topic: Set Operations

(v) Emp(eid,dno,sal) $\Pi_{eid}(Emp \bowtie_{(sal>s \wedge D=5} \rho_{I,D,S}(Emp))$ 100 emp && 10 from dept 5

Sol Ext: $\Pi_{eid}(Emp \bowtie_{sal>S} (\sigma_{D=5} (\rho_{I,D,S}(Emp))))$

100 tuples

100 tuples

Topic: Set Operations

(v) Emp(eid,dno,sal) $\Pi_{eid}(Emp \bowtie_{(sal>s \wedge D=5} \rho_{I,D,S}(Emp))$ 100 emp && 10 from dept 5

Sol Ext:

$$\Pi_{eid}(Emp \bowtie_{sal>S} (\sigma_{D=5} (\rho_{I,D,S}(Emp))))$$

100 tuples

100 comparisons

100 tuples

Topic: Set Operations

(v) Emp(eid,dno,sal) $\Pi_{eid}(Emp \bowtie_{(sal>s \wedge D=5} \rho_{I,D,S}(Emp))$ 100 emp && 10 from dept 5

Sol Ext:

$\Pi_{eid}(Emp \bowtie_{sal>S} (\sigma_{D=5} (\rho_{I,D,S}(Emp))))$

100 tuples

100 comparisons

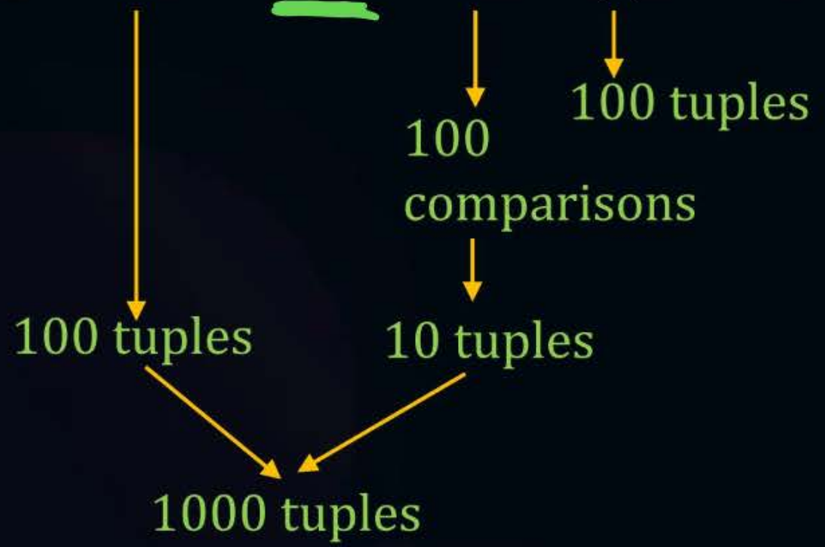
100 tuples

10 tuples

Topic: Set Operations

(v) Emp(eid,dno,sal) $\Pi_{eid}(Emp \bowtie_{(sal>s \wedge D=5} \rho_{I,D,S}(Emp))$ 100 emp && 10 from dept 5

Sol Ext: $\Pi_{eid}(Emp \bowtie_{sal>S} (\sigma_{D=5} (\rho_{I,D,S}(Emp))))$

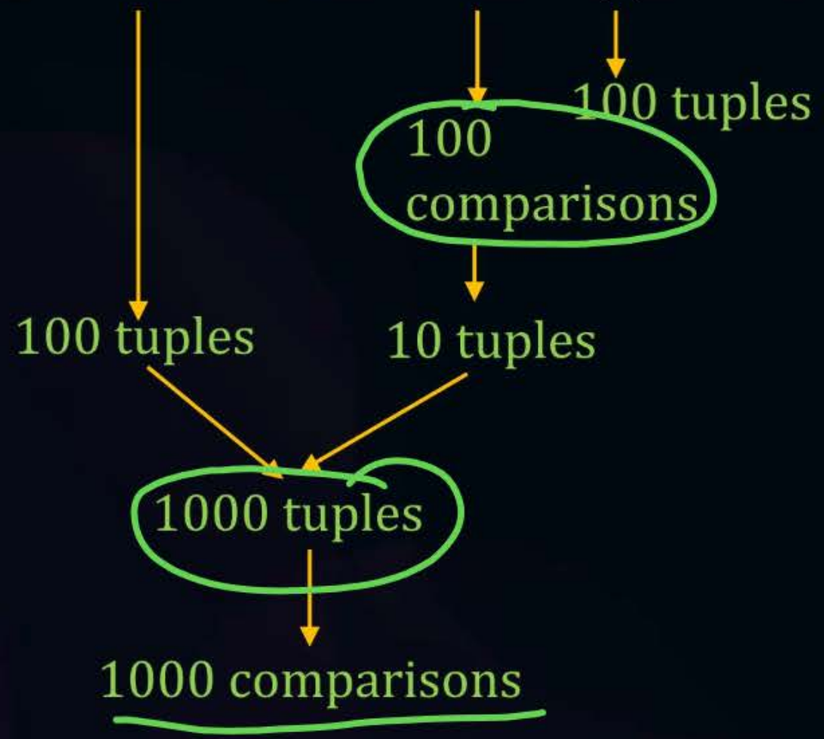


Topic: Set Operations

(v) Emp(eid,dno,sal) $\Pi_{eid}(Emp \bowtie_{(sal>s \wedge D=5} \rho_{I,D,S}(Emp))$ 100 emp && 10 from dept 5

Sol Ext:

$\Pi_{eid}(Emp \bowtie_{sal>S} (\sigma_{D=5} (\rho_{I,D,S}(Emp))))$



Topic: Set Operations

(v) Emp(eid,dno,sal) $\Pi_{eid}(Emp \bowtie_{(sal>s \wedge D=5} \rho_{I,D,S}(Emp))$ 100 emp && 10 from dept 5

Sol Ext:

$\Pi_{eid}(Emp \bowtie_{sal>S} (\sigma_{D=5} (\rho_{I,D,S}(Emp))))$

100 tuples

100 comparisons

100 tuples

10 tuples

1000 tuples

1000 comparisons

Cost of Comparisons : 1100 Comparisons from 20000 comparisons

Topic: Set Operations

I G S

#Q. Emp (eid, gen, sal)

$$\Pi_{eid}\,(Emp \bowtie_{gen=female \wedge G=male \wedge sal>S} (\rho_{I,G,S}(Emp)))$$

Topic: Set Operations

#Q. Emp (eid, gen, sal)

Π_{eid} (Emp$\bowtie_{gen=female \wedge G=male \wedge sal>S}(\rho_{I,G,S}(Emp))$

Sol: $\sigma_{gen=female}(Emp)$ $\sigma_{G=male}(\rho_{I,G,S}(Emp))$

Topic: Set Operations

#Q. Emp (eid, gen, sal)

Π_{eid} (Emp $\bowtie_{gen=female \wedge G=male \wedge sal>S}$ ($\rho_{I,G,S}$(Emp))

Sol: $\sigma_{gen=female}$(Emp) $\bowtie_{sal>S}$ $\sigma_{G=male}$ ($\rho_{I,G,S}$(Emp))

Topic: Set Operations

#Q. Emp (eid, gen, sal)

Π_{eid} (Emp$\bowtie_{gen=female \wedge G=male \wedge sal>S}(\rho_{I,G,S}(Emp))$

Sol: Π_{eid} ($\sigma_{gen=female}$(Emp)$\bowtie_{sal>S}\sigma_{G=male}$ ($\rho_{I,G,S}$(Emp)))

Topic: Queries

Queries related to Atleast/Only/Atmost:

Used for atleast 3 salaries

Atleast 4 employees

Only 3 salaries

Almost 3 employees

Topic: Queries

#Q. Enroll (Sid, Cid) Retrieve Sid's enrolled for atleast two courses

Enroll

Sid	Cid
S1	C1
S1	C2
S1	C3
S2	C1
S2	C2
S3	C1

atleast one course

Topic: Queries

#Q. Enroll (Sid, Cid) Retrieve Sid's enrolled for atleast two courses

Enroll

Sid	Cid
S1	C1
S1	C2
S1	C3
S2	C1
S2	C2
S3	C1

Sol.: For atleast 2 courses, consider the following logic, $t_1 \cdot Sid = t_2 \cdot Sid$ and $t_1 \cdot Cid \neq t_2 \cdot Cid$ where t_1, t_2 are two tuples of Relation Enroll.

Topic: Queries

#Q. Enroll (Sid, Cid) Retrieve Sid's enrolled for atleast two courses

Enroll

Sid	Cid
S1	C1
S1	C2
S1	C3
S2	C1
S2	C2
S3	C1

Enroll = T_1

Sid	Cid
S1	C1
S1	C2
S1	C3
S2	C1
S2	C2
S3	C1

Enroll = T_2

Sid	Cid
S1	C1
S1	C2
S1	C3
S2	C1
S2	C2
S3	C1

Sol.: For atleast 2 courses, consider the following logic, $t_1 \cdot Sid = t_2 \cdot Sid$ and $t_1 \cdot Cid \neq t_2 \cdot Cid$ where t_1, t_2 are two tuples of Relation Enroll.

Topic: Queries

#Q. Enroll (Sid, Cid) Retrieve Sid's enrolled for atleast two courses

Enroll

Sid	Cid
S1	C1
S1	C2
S1	C3
S2	C1
S2	C2
S3	C1

Enroll = T_1

Sid	Cid
S1	C1
S1	C2
S1	C3
S2	C1
S2	C2
S3	C1

Enroll = T_2

Sid	Cid
S1	C1
S1	C2
S1	C3
S2	C1
S2	C2
S3	C1

Sol.: For atleast 2 courses, consider the following logic, $t_1 \cdot Sid = t_2 \cdot Sid$ and $t_1 \cdot Cid \neq t_2 \cdot Cid$ where t_1, t_2 are two tuples of Relation Enroll.

$\rho(T_1, Enroll)$ $\rho(T_2, Enroll)$

Topic: Queries

#Q. Enroll (Sid, Cid) Retrieve Sid's enrolled for atleast two courses

Enroll

Sid	Cid
S1	C1
S1	C2
S1	C3
S2	C1
S2	C2
S3	C1

Enroll = T_1

Sid	Cid
S1	C1
S1	C2
S1	C3
S2	C1
S2	C2
S3	C1

Enroll = T_2

Sid	Cid
S1	C1
S1	C2
S1	C3
S2	C1
S2	C2
S3	C1

Sol.: For atleast 2 courses, consider the following logic, $t_1 \cdot Sid = t_2 \cdot Sid$ and $t_1 \cdot Cid \neq t_2 \cdot Cid$ where t_1, t_2 are two tuples of Relation Enroll.

$$\rho(T_1, Enroll) \bowtie_{T_1.Sid=T_2.Sid \wedge T_1.Cid \neq T_2.Cid} \rho(T_2, Enroll)$$

Topic: Queries

#Q. Enroll (Sid, Cid) Retrieve Sid's enrolled for atleast two courses

Enroll

Sid	Cid
S1	C1
S1	C2
S1	C3
S2	C1
S2	C2
S3	C1

Enroll = T_1

Sid	Cid
S1	C1
S1	C2
S1	C3
S2	C1
S2	C2
S3	C1

Enroll = T_2

Sid	Cid
S1	C1
S1	C2
S1	C3
S2	C1
S2	C2
S3	C1

Sol.: For atleast 2 courses, consider the following logic, $t_1 \cdot Sid = t_2 \cdot Sid$ and $t_1 \cdot Cid \neq t_2 \cdot Cid$ where t_1, t_2 are two tuples of Relation Enroll.

$$\Pi_{T_1.Sid}\ (\rho(T_1, Enroll) \bowtie_{T_1.Sid=T_2.Sid \wedge T_1.Cid \neq T_2.Cid} \rho(T_2, Enroll))$$

Topic: Queries

#Q. Enroll (Sid, Cid) Retrieve Sid's enrolled for atleast two courses

Enroll

Sid	Cid
S1	C1
S1	C2
S1	C3
S2	C1
S2	C2
S3	C1

Enroll = T_1

Sid	Cid
S1	C1
S1	C2
S1	C3
S2	C1
S2	C2
S3	C1

Enroll = T_2

Sid	Cid
S1	C1
S1	C2
S1	C3
S2	C1
S2	C2
S3	C1

Sol.: For atleast 2 courses, consider the following logic, $t_1 \cdot Sid = t_2 \cdot Sid$ and $t_1 \cdot Cid \neq t_2 \cdot Cid$ where t_1, t_2 are two tuples of Relation Enroll.

$$\Pi_{T_1.Sid}\left(\rho(T_1, Enroll) \bowtie_{T_1.Sid=T_2.Sid \wedge T_1.Cid \neq T_2.Cid} \rho(T_2, Enroll)\right)$$

or just rename the attribute names

Topic: Queries

#Q. Enroll (Sid, Cid) Retrieve Sid's enrolled for atleast two courses

Enroll

Sid	Cid
S1	C1
S1	C2
S1	C3
S2	C1
S2	C2
S3	C1

Enroll = T_1

Sid	Cid
S1	C1
S1	C2
S1	C3
S2	C1
S2	C2
S3	C1

Enroll = T_2

Sid	Cid
S1	C1
S1	C2
S1	C3
S2	C1
S2	C2
S3	C1

Sol.: For atleast 2 courses, consider the following logic, $t_1 \cdot Sid = t_2 \cdot Sid$ and $t_1 \cdot Cid \neq t_2 \cdot Cid$ where t_1, t_2 are two tuples of Relation Enroll.

$$\Pi_{T_1.Sid}\ (\rho(T_1, Enroll) \bowtie_{T_1.Sid=T_2.Sid \wedge T_1.Cid \neq T_2.Cid} \rho(T_2, Enroll))$$

or just rename the attribute names

$$\Pi_{Sid}\ ((Enroll) \bowtie_{Sid=S \wedge Cid \neq C} \rho_{S,\,C}(Enroll))$$

Topic: Queries

Questions that follows similar logic

- Employee working for atleast 2 projects
- Class that contains atleast 2 females

Topic: Queries

#Q. Retrieve Sid's that enrolled atleast 3 couries from Enroll (Sid, Cid)

Topic: Queries

#Q. Retrieve Sid's that enrolled atleast 3 couries from Enroll (Sid, Cid)

Sol.: Now, we have to consider 3 tuples t_1, t_2, t_3 and $t_1 \cdot Sid = t_2 \cdot Sid = t_3 \cdot Sid$ and t_1 Cid $\neq t_2 \cdot$ Cid, and $t_2 \cdot$ Cid $\neq t_3 \cdot$ Cid and $t_3 \cdot$ Cid $\neq t_1 \cdot$ Cid taking 3 instances of enroll i.e T_1, T_2, T_3

Topic: Queries

#Q. Retrieve Sid's that enrolled atleast 3 couries from Enroll (Sid, Cid)

Enroll = T_1

Sid	Cid
S1	C1
S1	C2
S1	C3
S2	C1
S2	C2
S3	C1

Enroll = T_2

Sid	Cid
S1	C1
S1	C2
S1	C3
S2	C1
S2	C2
S3	C1

Enroll = T_3

Sid	Cid
S1	C1
S1	C2
S1	C3
S2	C1
S2	C2
S3	C1

Sol.: Now, we have to consider 3 tuples t_1, t_2, t_3 and $t_1 \cdot Sid = t_2 \cdot Sid = t_3 \cdot Sid$ and t_1 Cid $\neq t_2 \cdot$ Cid, and $t_2 \cdot$ Cid $\neq t_3 \cdot$ Cid and $t_3 \cdot$ Cid $\neq t_1 \cdot$ Cid

taking 3 instances of enroll i.e T_1, T_2, T_3

Topic: Queries

#Q. Retrieve Sid's that enrolled atleast 3 couries from Enroll (Sid, Cid)

Enroll = T_1

Sid	Cid
S1	C1
S1	C2
S1	C3
S2	C1
S2	C2
S3	C1

Enroll = T_2

Sid	Cid
S1	C1
S1	C2
S1	C3
S2	C1
S2	C2
S3	C1

Enroll = T_3

Sid	Cid
S1	C1
S1	C2
S1	C3
S2	C1
S2	C2
S3	C1

Sol.: Now, we have to consider 3 tuples t_1, t_2, t_3 and $t_1 \cdot Sid = t_2 \cdot Sid = t_3 \cdot Sid$ and t_1 Cid $\neq t_2 \cdot$ Cid, and $t_2 \cdot$ Cid $\neq t_3 \cdot$ Cid and $t_3 \cdot$ Cid $\neq t_1 \cdot$ Cid

taking 3 instances of enroll i.e T_1, T_2, T_3

$$\rho(T_1, \text{Enroll}) \times \rho(T_2, \text{Enroll}) \times \rho(T_3, \text{Enroll})$$

Topic: Queries

#Q. Retrieve Sid's that enrolled atleast 3 couries from Enroll (Sid, Cid)

Enroll = T_1

Sid	Cid
S1	C1
S1	C2
S1	C3
S2	C1
S2	C2
S3	C1

Enroll = T_2

Sid	Cid
S1	C1
S1	C2
S1	C3
S2	C1
S2	C2
S3	C1

Enroll = T_3

Sid	Cid
S1	C1
S1	C2
S1	C3
S2	C1
S2	C2
S3	C1

Sol.: Now, we have to consider 3 tuples t_1, t_2, t_3 and $t_1 \cdot Sid = t_2 \cdot Sid = t_3 \cdot Sid$ and t_1 Cid $\neq t_2 \cdot$ Cid, and $t_2 \cdot$ Cid $\neq t_3 \cdot$ Cid and $t_3 \cdot$ Cid $\neq t_1 \cdot$ Cid

taking 3 instances of enroll i.e T_1, T_2, T_3

$$\Pi_{Sid}(\sigma_{(T_1 \cdot Sid = T_2 \cdot Sid \wedge T_2 \cdot Sid = T_3 \cdot Sid \wedge T_1 \cdot Cid \neq T_2 \cdot Cid \wedge T_2 \cdot Cid \neq T_3 \cdot Cid \wedge T_3 \cdot Cid \neq T_1 \cdot Cid)} \rho(T_1, Enroll) \times \rho(T_2, Enroll) \times \rho(T_3, Enroll)))$$

S1 C1 S1 C2 S1 C3

Topic: Queries

- In Relational Algebra, there is no other possible logic, if there query contains “Atleast 10” then we need to write logic for 10 tuples.
- But in SQL we have functions like min, max and count to do this.

Topic: Queries

#Q. Sid's enrolled only one Course

Enroll

Sid	Cid
S1	C1
S1	C2
S1	C3
S2	C1
S2	C2
S3	C1

Topic: Queries

#Q. Sid's enrolled only one Course

Sol.: We can solve this by the following logic.

{Sid's enrolled in only one course}

Enroll

Sid	Cid
S1	C1
S1	C2
S1	C3
S2	C1
S2	C2
S3	C1

Topic: Queries

#Q. Sid's enrolled only one Course

Sol.: We can solve this by the following logic.

$$\left\{\begin{array}{c}\text{Sid's enrolled} \\ \text{in only one} \\ \text{course}\end{array}\right\} = \left\{\begin{array}{c}\text{Sid's enrolled} \\ \text{in atleast one} \\ \text{course}\end{array}\right\}$$

Enroll

Sid	Cid
S1	C1
S1	C2
S1	C3
S2	C1
S2	C2
S3	C1

Topic: Queries

#Q. Sid's enrolled only one Course

Sol.: We can solve this by the following logic.

$$\{\text{Sid's enrolled in only one course}\} = \{\text{Sid's enrolled in atleast one course}\} - \{\text{Sid's enrolled atleast two courses}\}$$

Enroll

Sid	Cid
S1	C1
S1	C2
S1	C3
S2	C1
S2	C2
S3	C1

Topic: Queries

#Q. Sid's enrolled only one Course

Sol.: We can solve this by the following logic.

$$\left\{\begin{matrix}\text{Sid's enrolled}\\ \text{in only one}\\ \text{course}\end{matrix}\right\} = \left\{\begin{matrix}\text{Sid's enrolled}\\ \text{in atleast one}\\ \text{course}\end{matrix}\right\} - \left\{\begin{matrix}\text{Sid's enrolled}\\ \text{atleast two}\\ \text{courses}\end{matrix}\right\}$$

↓

The entire table

Enroll

Sid	Cid
S1	C1
S1	C2
S1	C3
S2	C1
S2	C2
S3	C1

Topic: Queries

#Q. Sid's enrolled only one Course

Sol.: We can solve this by the following logic.

$$\left\{\begin{array}{c}\text{Sid's enrolled}\\ \text{in only one}\\ \text{course}\end{array}\right\} = \left\{\begin{array}{c}\text{Sid's enrolled}\\ \text{in atleast one}\\ \text{course}\end{array}\right\} - \left\{\begin{array}{c}\text{Sid's enrolled}\\ \text{atleast two}\\ \text{courses}\end{array}\right\}$$

↓

The entire table

$$\text{Enroll} \bowtie_{\text{Sid}=S \wedge \text{Cid}\neq C} \rho_{S,C}(\text{Enroll})$$

Enroll S C

Sid	Cid
S1	C1
S1	C2
S1	C3
S2	C1
S2	C2
S3	C1

Topic: Queries

#Q. Sid's enrolled only one Course

Sol.: We can solve this by the following logic.

$$\left\{\begin{matrix}\text{Sid's enrolled} \\ \text{in only one} \\ \text{course}\end{matrix}\right\} = \left\{\begin{matrix}\text{Sid's enrolled} \\ \text{in atleast one} \\ \text{course}\end{matrix}\right\} - \left\{\begin{matrix}\text{Sid's enrolled} \\ \text{atleast two} \\ \text{courses}\end{matrix}\right\}$$

↓

The entire table

$$\Pi_{Sid}\,(Enroll) - \Pi_{Sid}\,(Enroll \bowtie_{Sid=S \wedge Cid \neq C}\ \rho_{S,C}(Enroll))$$

Enroll

Sid	Cid
S1	C1
S1	C2
S1	C3
S2	C1
S2	C2
S3	C1

Inspiring Stories : Chen Shu-Chu

Background: **Poor vegetable seller in Taiwan.**

Education: **Dropped out of school.**

Achievements: **Donated her life savings to build schools and libraries.**

Impact: **Showed true charity comes from the heart.**

Topic: Queries

#Q. Sid's enrolled only two courses

Enroll

Sid	Cid
S1	C1
S1	C2
S1	C3
S2	C1
S2	C2
S3	C1

Topic: Queries

#Q. Sid's enrolled only two courses

Sol.: Same logic

$$\{\text{Sid's enrolled in only two courses}\} = \{\text{Sid's enrolled in atleast two courses}\} - \{\text{Sid's enrolled in atleast three courses}\}$$

Enroll

Sid	Cid
S1	C1
S1	C2
S1	C3
S2	C1
S2	C2
S3	C1

Topic: Queries

#Q. Sid's enrolled only two courses

Sol.: Same logic

$$\left\{\begin{matrix}\text{Sid's enrolled in} \\ \text{only two} \\ \text{courses}\end{matrix}\right\} = \left\{\begin{matrix}\text{Sid's enrolled} \\ \text{in atleast two} \\ \text{courses}\end{matrix}\right\} - \left\{\begin{matrix}\text{Sid's enrolled} \\ \text{in atleast three} \\ \text{courses}\end{matrix}\right\}$$

S1
S2

$-$ | S1 | $=$ | S2 |

Enroll

Sid	Cid
S1	C1
S1	C2
S1	C3
S2	C1
S2	C2
S3	C1

Topic: Queries

#Q. Sid's enrolled only two courses

Sol.: Same logic

Enroll

Sid	Cid
S1	C1
S1	C2
S1	C3
S2	C1
S2	C2
S3	C1

$$\left\{\begin{array}{c}\text{Sid's enrolled in} \\ \text{only two} \\ \text{courie}\end{array}\right\} = \left\{\begin{array}{c}\text{Sid's enrolled} \\ \text{in atleast two} \\ \text{courses}\end{array}\right\} - \left\{\begin{array}{c}\text{Sid's enrolled} \\ \text{in atleast three} \\ \text{couries}\end{array}\right\}$$

S1
S2

$-$ | S1 | $=$ | S2 |

$$= \Pi_{Sid}\ (Enrolll \bowtie_{Sid=S \wedge Cid \neq C}\ \rho_{S,\,C}(Enroll))$$

$$\Pi(\sigma_{\substack{T_1 \cdot Sid = T_2 \cdot Sid \wedge T_2 \cdot Sid \\ = T_3 \cdot Sid \wedge T_1 \cdot Cid \\ \neq T_2 \cdot Cid \wedge T_2 \cdot Cid \\ \neq T_3 \cdot Cid \wedge T_3 \cdot Cid \\ \neq T_1 \times Cid}}(\rho(T_1, Enroll) \times \rho(T_2, Enroll) \times \rho(T_3, Enroll)))$$

Topic: Queries

#Q. Sid's enrolled in atmost one course in the Relation

0, 1

Enroll

Sid	Cid
S1	C1
S1	C2
S1	C3
S2	C1
S2	C2
S3	C1

Topic: Queries

#Q. Sid's enrolled in atmost one course in the Relation

All Students

Stud

Sid	Sname
S1	
S2	
S3	
S4	

Enroll

Sid	Cid
S1	C1
S1	C2
S1	C3
S2	C1
S2	C2
S3	C1

S4 ✓

S3 S4

Topic: Queries

#Q. Sid's enrolled in atmost one course in the Relation

Sol.: $\left\{\begin{matrix}\text{Sid's enrolled in} \\ \text{atmost one} \\ \text{course}\end{matrix}\right\} = \{\text{All students}\} - \left\{\begin{matrix}\text{Sid's enrolled} \\ \text{in atleast two} \\ \text{courses}\end{matrix}\right\}$

All Students

Stud

Sid	Sname
S1	
S2	
S3	
S4	

Enroll

Sid	Cid
S1	C1
S1	C2
S1	C3
S2	C1
S2	C2
S3	C1

Topic: Queries

#Q. Sid's enrolled in atmost one course in the Relation

Sol.: $\left\{\begin{matrix}\text{Sid's enrolled in} \\ \text{atmost one} \\ \text{course}\end{matrix}\right\} = \{\text{All students}\} - \left\{\begin{matrix}\text{Sid's enrolled} \\ \text{in atleast two} \\ \text{courses}\end{matrix}\right\}$

$= \Pi_{Sid}(Stud) - \Pi_{Sid}(Enroll \bowtie_{Sid=S \wedge Cid \neq C} \rho_{S,C}(Enroll))$

All Students

Stud

Sid	Sname
S1	
S2	
S3	
S4	

Enroll

Sid	Cid
S1	C1
S1	C2
S1	C3
S2	C1
S2	C2
S3	C1

Topic: Queries

0, 1, 2, 3, 4, 5, 6

#Q. Sid's enrolled in atmost two courses

Topic: Queries

#Q. Sid's enrolled in atmost two courses

Sol.: = All students – { Sid's enrolled in atleast three courses }

All Students

Stud

Sid	Sname
S1	
S2	
S3	
S4	

Enroll

Sid	Cid
S1	C1
S1	C2
S1	C3
S2	C1
S2	C2
S3	C1

Topic: Queries

#Q. Sid's enrolled in atmost two courses

Sol.: = All students – { Sid's enrolled in atleast three courses }

⇒ If stud is not given then atleast one and atmost one have same queries

All Students

Stud

Sid	Sname
S1	
S2	
S3	
S4	

Enroll

Sid	Cid
S1	C1
S1	C2
S1	C3
S2	C1
S2	C2
S3	C1

Topic: Queries

Exploring Division Queries:

#Q. Given Relation R with X set of attributes and n distinct tuples and a Relation S with y set of attributes and m distict tuples

for $\Pi_{\underbrace{AB}_{x}}(R)/\Pi_{\underbrace{B}_{y}}(s)$ then ?

(i) Condition Required to use R/S (for Division Op)

(ii) Resulted attribute set of R/S is?

(iii) Cordinality of R/S? min & max?

Topic: Queries

Exploring Division Queries:

#Q. Given Relation R with X set of attributes and n distinct tuples and a Relation S with y set of attributes and m distict tuples

for $\Pi_{\underbrace{AB}_{x}}(R)/\Pi_{\underbrace{B}_{y}}(s)$ then ?

$\pi_X(R)/\pi_Y(R)$

$X \supset Y$

(i) Condition Required to use R/S (for Division Op) Sol: (i) X > Y

(ii) Resulted attribute set of R/S is?

(iii) Cordinality of R/S? min & max?

Topic: Queries

Exploring Division Queries:

#Q. Given Relation R with X set of attributes and n distinct tuples and a Relation S with y set of attributes and m distict tuples

for $\Pi_{\underbrace{AB}_{x}}(R)/\Pi_{\underbrace{B}_{y}}(s)$ then ?

(i) Condition Required to use R/S (for Division Op) Sol: (i) X > Y

(ii) Resulted attribute set of R/S is? Sol: X - Y

(iii) Cordinality of R/S? min & max?

Topic: Queries

Exploring Division Queries:

#Q. Given Relation R with X set of attributes and n distinct tuples and a Relation S with y set of attributes and m distict tuples

for $\Pi_{\underbrace{AB}_{x}}(R)/\Pi_{\underbrace{B}_{y}}(s)$ then ?

(i) Condition Required to use R/S (for Division Op) Sol: (i) X > Y

(ii) Resulted attribute set of R/S is? Sol: X - Y

Topic: Queries

Sol.: (iii) Cordinality of R/S? min & max?

0

R

A	B
S1	C1
S1	C2
S1	C3
S2	C1
S2	C2
S3	C1

n

S

B
b1
b2
b3

m

Topic: Queries

Sol.: (iii) Cordinality of R/S? min & max?

min → when no A values is mapped with all B values

R

A	B
S1	C1
S1	C2
S1	C3
S2	C1
S2	C2
S3	C1

n

S

B
b1
b2
b3

m

Topic: Queries

Sol.: (iii) Cordinality of R/S? min & max?

min → when no A values is mapped with all B values

∴ 0 tuples

R

A	B
S1	C1
S1	C2
S1	C3
S2	C1
S2	C2
S3	C1

n

S

B
b1
b2
b3

m

Topic: Queries

Sol.: (iii) Cordinality of R/S? min & max?

min → when no A values is mapped with all B values

∴ 0 tuples

max → All distinct values of A is mapped with all unique values of B

R

A	B
S1	b1
S1	b2
S1	b3
S2	b1
S2	b2
S3	b1

n

S

B
b1
b2
b3

m

Topic: Queries

Sol.: (iii) Cordinality of R/S? min & max?

min → when no A values is mapped with all B values

∴ 0 tuples

max → All distinct values of A is mapped with all unique values of B

= no. of distinct values of A

R

A	B
S1	C1
S1	C2
S1	C3
S2	C1
S2	C2
S3	C1

n

S

B
b1
b2
b3

m

Topic: Queries

Sol.: (iii) Cordinality of R/S? min & max?

R

A	B
S1	C1
S1	C2
S1	C3
S2	C1
S2	C2
S3	C1

n

S

B
b1
b2
b3

m

$\lfloor n/m \rfloor$

min → when no A values is mapped with all B values

∴ 0 tuples

max → All distinct values of A is mapped with all unique values of B

= no. of distinct values of A

= $\lfloor n \mid m \rfloor$ $\lfloor n/m \rfloor$

Topic: Queries

Sol.: (iii) Cordinality of R/S? min & max?

R

A	B
S1	C1
S1	C2
S1	C3
S2	C1
S2	C2
S3	C1

n

S

B
b1
b2
b3

m

min → when no A values is mapped with all B values

∴ 0 tuples

max → All distinct values of A is mapped with all unique values of B

= no. of distinct values of A

= $\lfloor n \mid m \rfloor$

Assuming $m > 0$

Inspiring Stories : Harbans Lal Mehta

Background: **Born in Punjab, faced Partition struggles.**

Education: **Studied with great effort.**

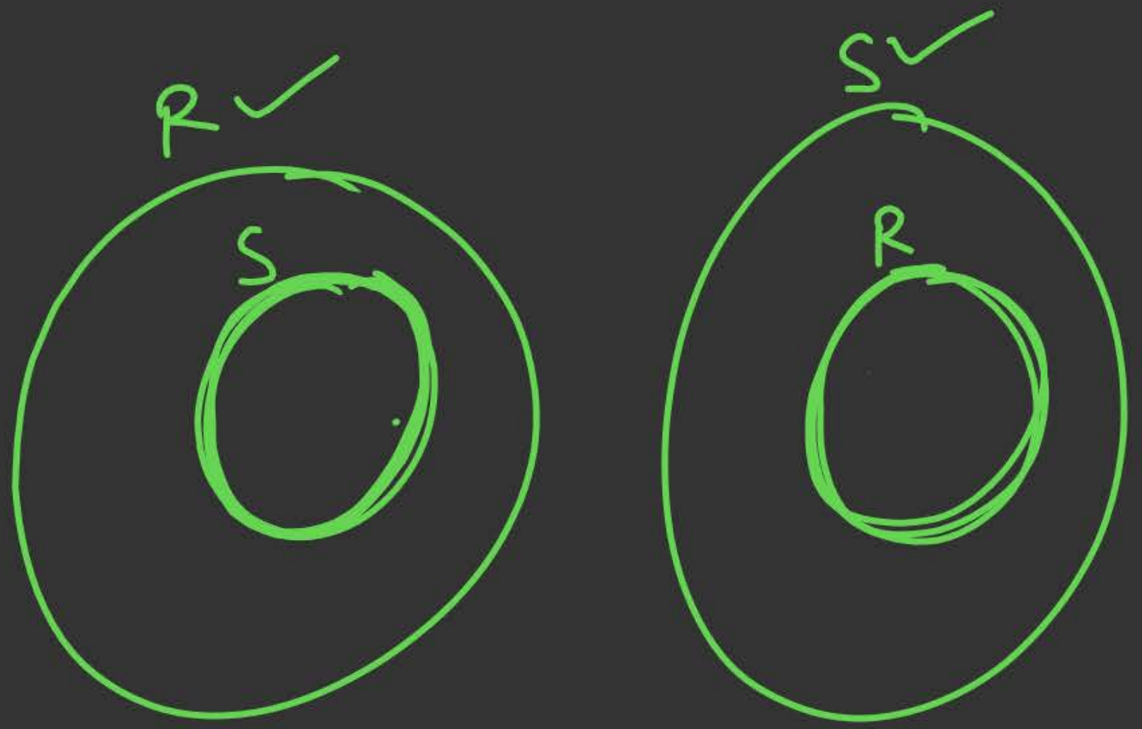
Achievements: **Created affordable healthcare programs in villages.**

Impact: **Saved thousands of rural lives.**

Topic: Queries

⇒ For a Relation R with n distinct tuples and Relation S with m distinct tuples

	RA Expression	Cordiality (min, max)	Commulative
1	$\pi_A R$	Atmost n tuples (Unique values of A)	No
2	$\sigma_C(R)$	(0, n)	Yes
3	$R \times S$	n × m tuples	Yes
4	$R \bowtie S$, $R \bowtie_C S$	(0, n * m) tuples	Yes
5	$R \rtimes S$	(n, n * m) tuples	No
6	$R \bowtie S$	(m, n * m) tuples	No
7	$R \rtimes S$	(max(n, m), n * m)	Yes
8	$R \cup S$	(max (n, m), n + m)	Yes
9	$R \cap S$	(0, min (n, m))	Yes
10	$R - S$	(0, n)	No
11	R/S	$(0, \lfloor n \mid m \rfloor)$	No


R
S
S
R

Topic: Queries

#Q. Stud (Sid, sname) sid is primary key with 100 tuples

Enroll (Sid Cid) sid Cid is primary key with 50 tuples

How many (min, max) tuples in result of stud ⋈ Enroll ?

(a) (100, 50) (b) (100, 0) (c) (50, 50) (d) (5000, 0)

Stud

Sid	Sname
S1	A
S2	A
S3	B
S4	B
S5	B

Enroll

Sid	Cid
S1	C1
S1	C2
S1	C3
S2	C3

Only once

Topic: Queries

#Q. Stud (Sid, sname) sid is primary key with 100 tuples

Enroll (Sid Cid) sid Cid is primary key with 50 tuples

How many (min, max) tuples in result of stud ⋈ Enroll ?

(a) (100, 50) (b) (100, 0) (c) (50, 50) (d) (5000, 0)

Sol: The max possible are 50

Stud

Sid	Sname
S1	A
S2	A
S3	B
S4	B
S5	B

Enroll

Sid	Cid
S1	C1
S1	C2
S1	C3
S2	C3

Only once

Topic: Queries

#Q. Stud (Sid, sname) sid is primary key with 100 tuples

Enroll (Sid Cid) sid Cid is primary key with 50 tuples

How many (min, max) tuples in result of stud ⋈ Enroll ?

(a) (100, 50) (b) (100, 0) (c) (50, 50) (d) (5000, 0)

Sol: The max possible are 50

min possible =50 (Assumption Enroll. Sid refers to student. Sid

There is no Sid in Enroll which is not present in stud table)

Stud

Sid	Sname
S1	A
S2	A
S3	B
S4	B
S5	B

Enroll

Sid	Cid
S1	C1
S1	C2
S1	C3
S2	C3

50

Only once

Topic: Queries

#Q. Stud (Sid, sname) sid is primary key with 100 tuples

Enroll (Sid Cid) sid Cid is primary key with 50 tuples

How many (min, max) tuples in result of stud ⋈ Enroll ?

(a) (100, 50) (b) (100, 0) (c) (50, 50) (d) (5000, 0)

Sol: The max possible are 50

min possible =50 (Assumption Enroll. Sid refers to student. Sid

There is no Sid in Enroll which is not present in stud table)

= 0 if no Foreign key exists(If stud.Sid = Enroll.Sid does not satisfy)

(50, 50) since it is present in option

PK

Stud

Sid	Sname
S1	A
S2	A
S3	B
S4	B
S5	B

FK

Enroll

Sid	Cid
S1	C1
S1	C2
S1	C3
S2	C3

50

Only once

Topic: Queries

#Q. R(A B C) -> (No Nulls with n tuples)

S(BDE) -> m tuples

⇒ R.B is Non key and S.B is Key

How many tuples will be present in R ⋈ S?

max = _____

min = _____

Topic: Queries

Sol.:

R

A	B	C
A1	2	
a2	2	
a3	2	
a4	3	
a5	3	

n

⇑ Nonkey

S

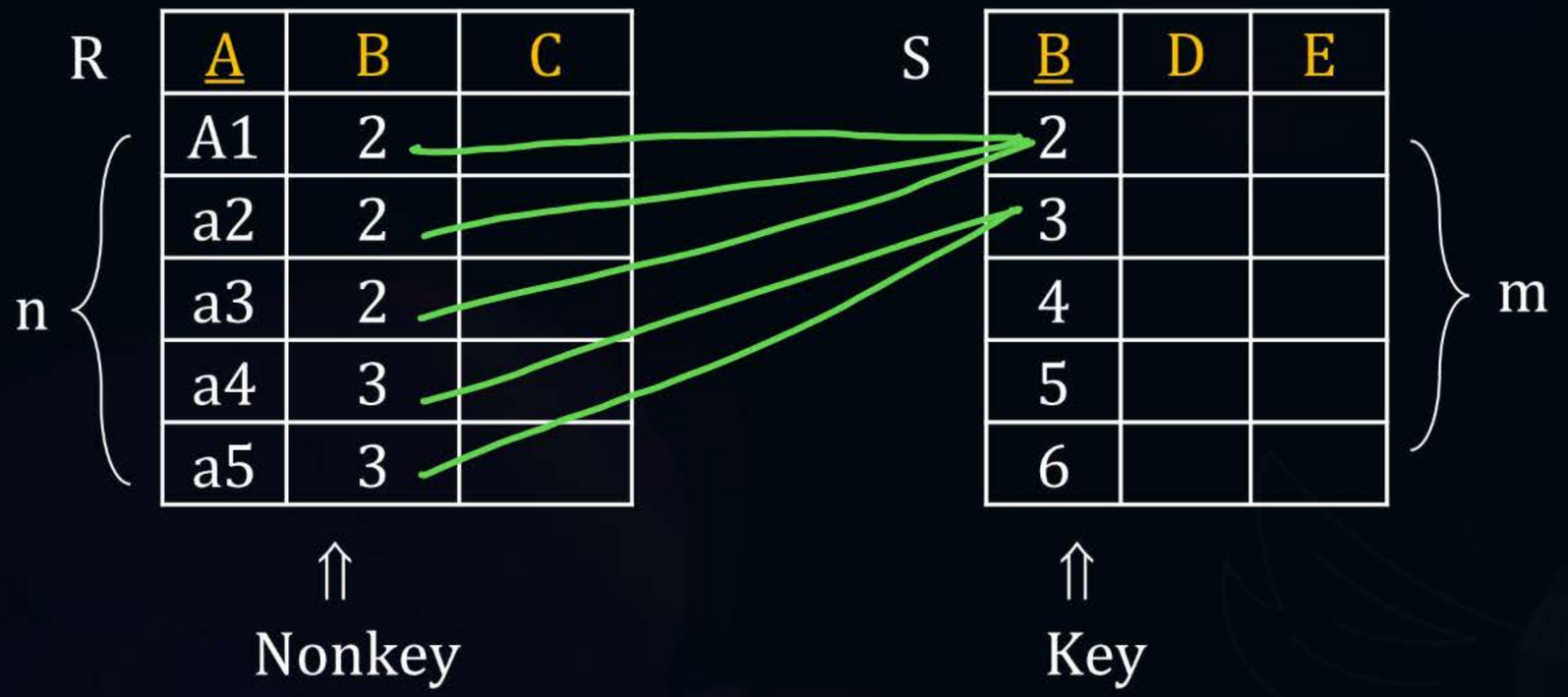
B	D	E
2		
3		
4		
5		
6		

m

⇑ Key

Topic: Queries

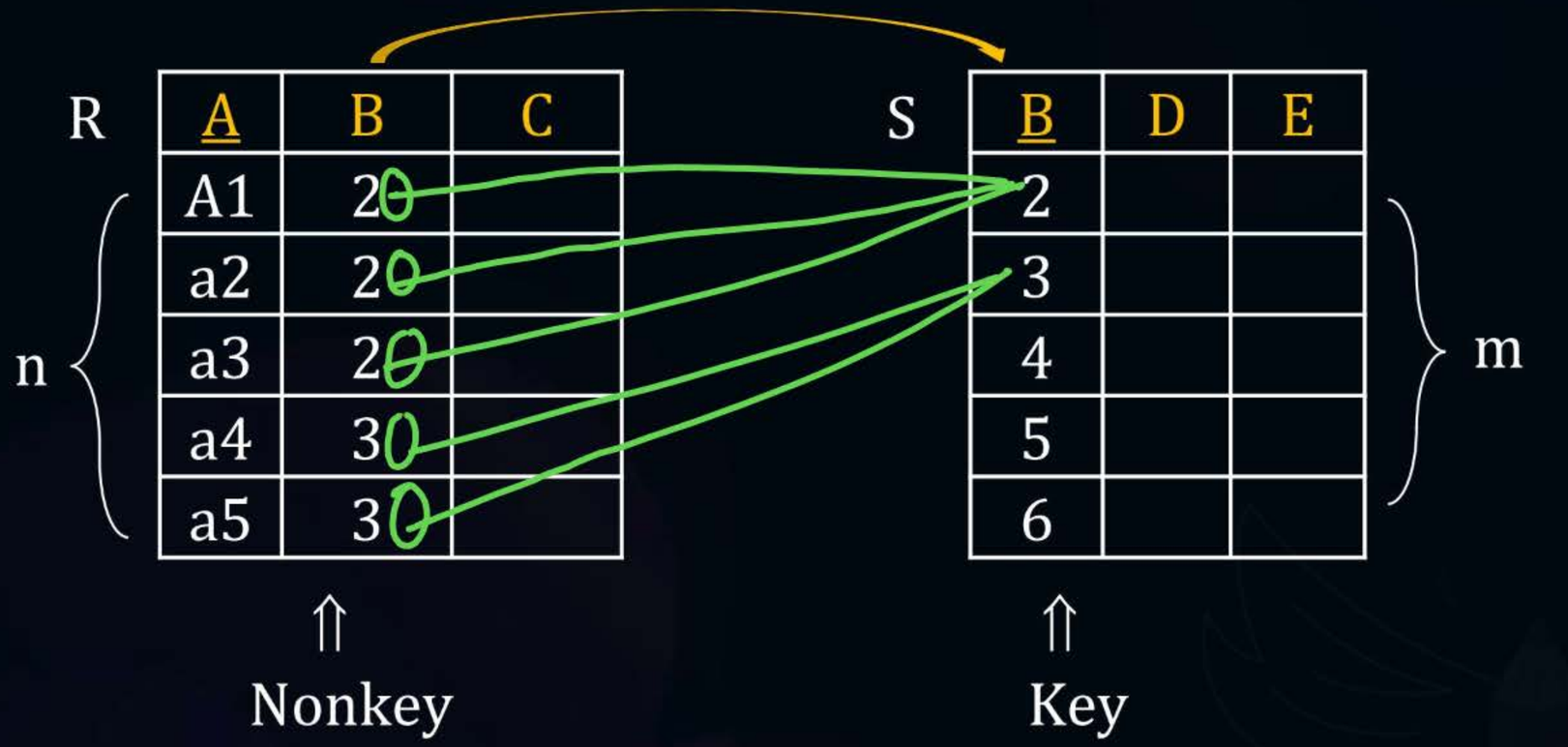
Sol.:



max = n tuples

Topic: Queries

Sol.:



max = n tuples

min = n tuples (with FK)

0 tuples (with No FK)

Topic: Queries

(2) R(<u>A</u>BC) S(<u>A</u>DE) ⇒ Key and key

↓ ↓

n tuples m tuples

max tuples in result of R ⋈ S ?

min tuples in result of R ⋈ S ?

R

A	B	C
2		
3		
4		
5		
6		

⇑ K

S

A	D	E
2		
3		
4		
5		

⇑ K

Topic: Queries

$\therefore$ max = min (n, m)

$$\text{min} = \begin{cases} \text{min } (n, m) & \text{with FK} \\ 0 & \text{with No FK} \end{cases}$$

PK

FK

R

A	B	C
2		
3		
4		
5		
6		

S

A	D	E
2		
3		
4		
5		

⇑ K

⇑ K

Topic: Queries

(3) R(A B C) S(D B E) ⇒ NKey and Nkey

n tuples m tuples

max tuples in result of R ⋈ S : _____

min tuples in result of R ⋈ S : _____

R

A	B	C
a1	2	
a2	2	
a3	2	

⇑
NK

S

D	B	E
a1	2	
a2	2	

⇑
NK

Topic: Queries

Max = n * m

A	B	C
a1	2	
a2	2	
a3	2	

D	B	E
a1	2	
a2	2	

NK

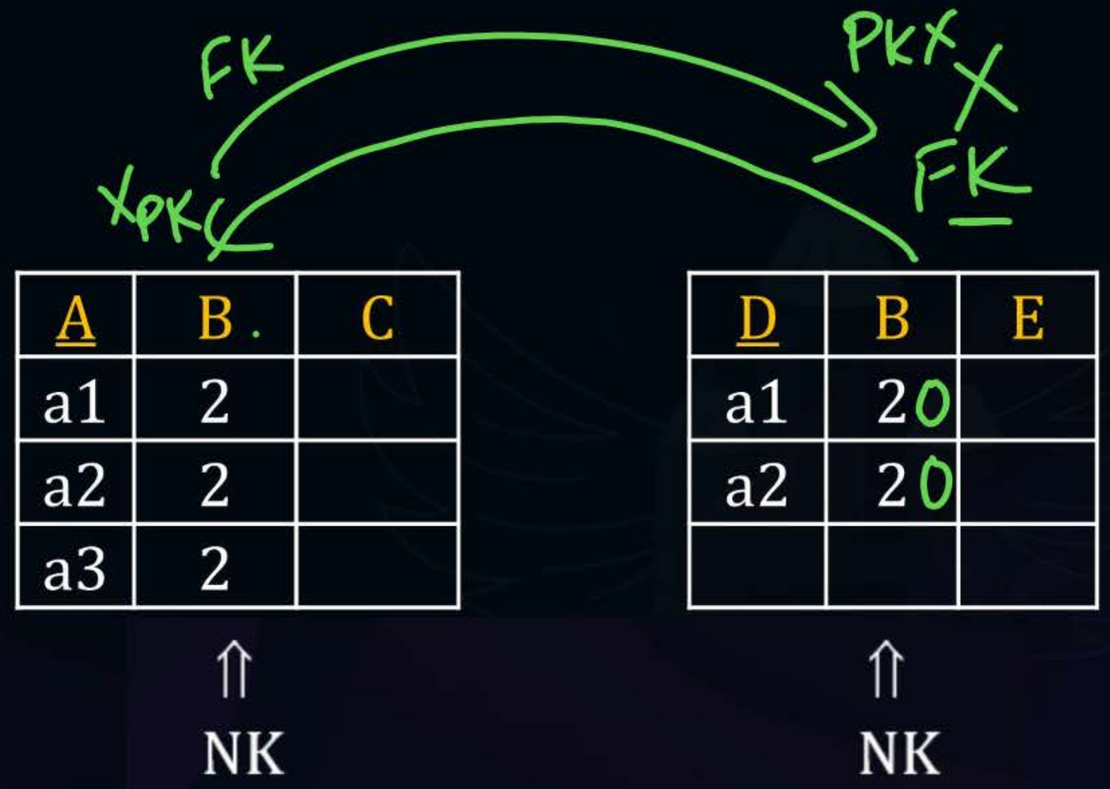
NK

Topic: Queries

Max = n * m

min = 0

because no possibility B being a foreign key



2 mins Summary

Relation Algebra → 4 marks

Very confusing

So requires practice

This is prerequisite for TRC

Telegram channel

@RBRSIRPW

THANK - YOU