COMPUTER SCIENCE

Database Management System

Query Language

Lecture_1

Vijay Agarwal sir

TOPICS TO BE COVERED

01 Relational Algebra

02 Operators

1. FD & Normalization
2. Transaction & Concurrency Control
3. ER MODEL & Conversion & foreign key Concept
4. Query language

Relational - Model (RDBMS)

E.F. Codd

Theoritical : RA, RC

Practical : SQL

Retrive

Query Language

① Procedural Q.L

↳ Relational Algebra.

② Non Procedural Q.L

↳ SQL

↳ (TRC & DRC) RC

Query Language

Procedural Query Language

- → WHAT to Reterive from DB
- → HOW to Reterive from DB

Relational Algebra:

Non Procedural Query Language

- → WHAT to Reterive from DB.

- ❑ SQL
- ❑ TRC

Procedural Query Language and Non-procedural Query Language

Procedural Query Language	Non-procedural Query Language
Formulation of how to access data from the database table and what data required to retrieve from DB tables. "Relational Algebra"	Formulation of what data retrieve from DB tables. "Relational Calculus" "SQL"

Relational Algebra.

The Basic Idea of Query language is Query executed on a DB Table (Relation) Tuple by Tuple (Step By Step) (Row by Row)

One Tuple at a time.

Difference b/w R.A & SQL.

RA

By default
↳ Distinct
output

SQL

By default
↳ Duplicate Retain.

Relational Algebra

(Always generate distinct tuples)

Relational algebra refers to a procedural query language that takes relation instances as input and returns relation instances as output

Input : Relation (Table)

Output : Relation (Table)

Relational Algebra

Basic operators

1. Selection [σ]
2. Projection [π]
3. CROSS Product [X]
4. UNION [$\cup$]
5. Set Difference [$-$]
6. Rename [ρ]

Derived operator

1. JOIN ($\bowtie$) & its type (Using X, σ, π)
2. Division [/]
3. Intersection [$\cap$]

$$\boxed{R \cap S = R - (R - S)}$$

$$\boxed{R \cap S = R - (R - S)}$$

L.H.S

$R \cap S = [4, 5, 6]$

eg) $R : [1, 2, 3, 4, 5, 6]$

$S : [4, 5, 6, 7, 8, 9, 10, 11]$

R.H.S $\Rightarrow \overset{R-S}{[1, 2, 3]}$

$R - (R - S) \Rightarrow [1, 2, 3, 4, 5, 6] - [1, 2, 3] = [4, 5, 6] \equiv R \cap S$

Relational Algebra

Basic operators

1. π : Projection operator
2. σ : Selection operator
3. $\times$: Cross-product operator
4. $\cup$: Union
5. $-$: Set difference / MINUS
6. ρ : Rename operator [ρ]

Relational Algebra

Derived operators

1. $\cap$: Intersection {using "–"}
2. $\bowtie$: Join {using X, σ} & its type
3. / or $\div$: Division {using π, x, –}

Selection [σ] :

It Select the Tuples / Record/Row Based on specified Condition.

Syntax

$$\boxed{\sigma_{Condition}(Relation)}$$

eg

$\sigma_{CGPA>9}(Student)$

$\sigma_{State='WB'}(Student)$

Projection [Π] : It Select field/Attribute/Attribute list from Relation.

Syntax

$$\boxed{\begin{array}{c}\Pi_{Attribute} \\ \text{or} \\ \Pi_{Attribute\ list}\end{array}(Relation)}$$

eg

$\Pi_{sname}(Student)$

$\Pi_{sname,Contact}(Student)$

⋮

① $\pi_{RollNo}(Student)$

O/P →

RollNo
1
2
3
4
5
6
7

STUDENT

RollNo	Name	CGPA
1	Abhay	9
2	Priyanshu	10
3	Shivansh	9
4	Sourav.	10
5	Priyanshu	9
6	Tarun	10
7	Abhay	9

② $\pi_{Name}(STUDENT)$

O/P →

Abhay
Priyanshu
Shivansh
Sourav
Tarun.

STUDENT

RollNo	Name	CGPA
1	Abhay	9
2	Priyanshu	10
3	Shivansh	9
4	Sourav.	10
5	Priyanshu	9
6	Tarun	10
7	Abhay	9

① $\sigma_{CGPA>9}(Student)$

RollNo	Name	CGPA
2	Priyanshu	10
4	Sourave	10
6	Tarun	10

② $\Pi_{Name}[\sigma_{CGPA>9}(Student)]$

O/P

Name
Priyanshu
Sourav.
Tarun

Relational Algebra

PW

Basic operators

field | column

I. π : Projection [π]

- $\pi_{\text{Attribute name}}$ (R): It is used to project required attribute from relation R.

Tuple | Row

- $\sigma_{\text{Condition(P)}}$ (R): It is used to select records from relation R, those satisfied the condition (P).

$\sigma_{\text{condition}}(R)$

Q) W.A.Q (Write a Query) to find Pname of Red Color Parts ?

Parts

Pid	Pname	Color
1	A	Red
2	B	Green
3	C	Red
4	D	Green
5	A	yellow
6	C	Blue.

Soln)

$$\pi_{Pname}\left(\sigma_{Color=Red}(Parts)\right)$$

Pname
A
C

Ans

Example:

R

A	B	C
8	4	5
2	4	5
7	4	6
3	5	5

① $\pi_{B,C}(R)$ → O/P

$\pi_{B,C}(R)$:

B	C
4	5
4	6
5	5

②

$\sigma_{A>5}(R)$:

A	B	C
8	4	5
7	4	6

1 Reserves (R_1)

Sid	Bid	day
22	101	10/10/96
58	103	11/12/96

2 Sailors(S_1)

Sid	Sname	Rating	age
22	dustin	7	45.0
31	lubber	8	55.5
58	rusty	10	35.0

3 Sailors(S_2)

Sid	Sname	Rating	age
28	yuppy	9	35.0
31	lubber	8	55.5
44	guppy	5	35.0
58	rusty	10	35.0

Selection (σ)

R.A ⇒ Maintain the Order.

Q.1 $\sigma_{\text{rating} > 8}(S_2)$

Sid	Sname	Rating	Age
28	yuppy	9	35.0
58	Rusty	10	35.0

Sailors (S_2)

Sid	Sname	Rating	age
28	yuppy	9	35.0
31	lubber	8	55.5
44	guppy	5	35.0
58	rusty	10	35.0

Selection (σ)

Ans.1 $\sigma_{\text{rating} > 8}(S_2)$

Output:

Sid	Sname	Rating	age
28	yuppy	9	35.0
58	rusty	10	35.0

Sailors (R_2)

Sid	Sname	Rating	age
28	yuppy	9	35.0
31	lubber	8	55.5
44	guppy	5	35.0
58	rusty	10	35.0

Q. $\pi_{sname,\ rating}\ (\sigma_{rating > 8}\ (S_2))$

Output:

Sname	Rating
yuppy	9
rusty	10

Sailors(S_2)

Sid	Sname	Rating	age
28	yuppy	9	35.0
31	lubber	8	55.5
44	guppy	5	35.0
58	rusty	10	35.0

Q. $\underbrace{\pi_{sname,\ rating}}_{II^{nd}} (\underbrace{\sigma_{rating > 8}}_{I^{st}} (S_2))$

Output:

Sname	Rating
yuppy	9
rusty	10

Sailors(S_2)

Sid	Sname	Rating	age
28	yuppy	9	35.0
~~31~~	~~lubber~~	~~8~~	~~55.5~~
~~44~~	~~guppy~~	~~5~~	~~35.0~~
58	rusty	10	35.0

Projection(π)

Q.2 $\pi_{age}(S_2)$

Age
35.0
55.5

Sailors (R_2)

Sid	Sname	Rating	age
28	yuppy	9	35.0
31	lubber	8	55.5
44	guppy	5	35.0
58	rusty	10	35.0

Projection(π)

Ans.2 $\pi_{age}(S_2)$

Sailors (S_2)

Sid	Sname	Rating	age
28	yuppy	9	35.0
31	lubber	8	55.5
44	guppy	5	35.0
58	rusty	10	35.0

Q)

$\pi_{sname\ Rating}(S_2)$

Output:

age
35.0
55.5

Q. $\pi_{\text{sname, rating}}(S_2)$

Output:

Sname	Rating
yuppy	9
lubber	8
guppy	5
rusty	10

Some Important Points / Conculsion .

1 Reserves (R_1)

Sid	Bid	day
22	101	10/10/96
58	103	11/12/96

2 Sailors(S_1)

Sid	Sname	Rating	age
22	dustin	7	45.0
31	lubber	8	55.5
58	rusty	10	35.0

3 Sailors(S_2)

Sid	Sname	Rating	age
28	yuppy	9	35.0
31	lubber	8	55.5
44	guppy	5	35.0
58	rusty	10	35.0

Sid	Bid	day
22	101	10/10/96
58	103	11/12/96

1st C_1 1st C_2

Sailors(S_1)

Sid	Sname	Rating	age
22	dustin	7	45.0
31	lubber	8	55.5
58	rusty	10	35.0

Sailors(S_2)

Sid	Sname	Rating	age
28	yuppy	9	35.0
31	lubber	8	55.5
44	guppy	5	35.0
58	rusty	10	35.0

$$\sigma_{C_3}(\sigma_{C_2}(\sigma_{C_1}(R))) \equiv \sigma_{C_2}(\sigma_{C_1}(\sigma_{C_3}(R)))$$

C_1: Rating > 9

C_2: Age ≥ 35.

C_1: Condition1

C_2: Condition2

C_3: Condition3

$$\sigma_{C_1}(\sigma_{C_2}(S_1)) \Rightarrow \underbrace{\sigma_{Rating > 9}}_{\text{II}^{nd}}\left(\underbrace{\sigma_{Age \geq 35}(S_1)}_{\text{I}^{st}}\right)$$

o/p

> 9

58	rusty	10	35.0

45.0
55.5
35.0

3 Tuple complete

1 Reserves (R_1)

Sid	Bid	day
22	101	10/10/96
58	103	11/12/96

2 Sailors(S_1)

9

Sid	Sname	Rating	age
22	dustin	7	45.0
31	lubber	8	55.5
58	rusty	10	35.0

3 Sailors(S_2)

Sid	Sname	Rating	age
28	yuppy	9	35.0
31	lubber	8	55.5
44	guppy	5	35.0
58	rusty	10	35.0

$$\boxed{\sigma_{C_3}(\sigma_{C_2}(\sigma_{C_1}(R))) \equiv \sigma_{C_2}(\sigma_{C_1}(\sigma_{C_3}(R)))}$$

C_1: Rating > 9

C_2: Age ≥ 35.

C_1: Condition1

C_2: Condition2

C_3: Condition3.

$$\sigma_{C_2}(\sigma_{C_1}(R)) \Rightarrow \underbrace{\sigma_{Age \geq 35}}_{2^{nd}}\left(\underbrace{\sigma_{Rating > 9}}_{1^{st}}(S_1)\right)$$

10

o/p

$$\boxed{58 \quad \text{rusty} \quad 10 \quad 35.0}$$

1. $$\boxed{\sigma_{C_2}(\sigma_{C_1}(R)) \equiv \sigma_{C_1}(\sigma_{C_2}(R))}$$

Selection [σ] is cummutative.

'AND'

2. $$\boxed{\sigma_{C_2}(\sigma_{C_1}(R)) \equiv \sigma_{C_1 \wedge C_2}(R)}$$

3. $$\sigma_{Age \geq 35}(\sigma_{Rating > 9}(S_1)) \equiv \sigma_{Age \geq 35 \wedge Rating > 9}(S_1)$$

1 Reserves (R_1)

Sid	Bid	day
22	101	10/10/96
58	103	11/12/96

2 Sailors(S_1)

Sid	Sname	Rating	age
22	dustin	7	45.0
31	lubber	8	55.5
58	rusty	10	35.0

3 Sailors(S_2)

Sid	Sname	Rating	age
28	yuppy	9	35.0
31	lubber	8	55.5
44	guppy	5	35.0
58	rusty	10	35.0

③ $$\boxed{\underset{\text{L.H.S}}{\pi_{A_1}\left[\sigma_{C_1}(R)\right]} \neq \underset{\text{R.H.S}}{\sigma_{C_1}\left[\pi_{A_1}(R)\right]}}$$

Condition

C_1 : Rating > 9

A_1 (Attribute) : Name

L.H.S

$$\underbrace{\pi_{Name}}_{2^{nd}}\left[\underbrace{\sigma_{Rating>9}(S_1)}_{1^{st}}\right]$$

o/p →

Name
Rusty

$$\underbrace{\sigma_{Rating \geq 9}}_{II^{nd}}\left(\underbrace{\pi_{Name}(S_1)}_{I^{st}}\right)$$

⇓ o/p

$$\sigma_{Rating \geq 9}\left[\begin{array}{|c|} \hline \text{dustin} \\ \text{lubber} \\ \text{rusty} \\ \hline \end{array}\right]$$

o/p → Empty

① Reserves (R_1)

Sid	Bid	day
22	101	10/10/96
58	103	11/12/96

② Sailors(S_1)

Sid	Sname	Rating	age
22	dustin	7	45.0
31	lubber	8	55.5
58	rusty	10	35.0

③ Sailors(S_2)

Sid	Sname	Rating	age
28	yuppy	9	35.0
31	lubber	8	55.5
44	guppy	5	35.0
58	rusty	10	35.0

Q In this eg. L.H.S Can be Replaced / Transformed by R.H.S

(4) $$\sigma_{C_1}\left[\pi_{A_1}(R)\right] \longrightarrow \pi_{A_1}\left[\sigma_{C_1}(R)\right]$$

(L.H.S) (R.H.S)

YES ✓

In this (L.H.S Parts) firstly Attribute A_1 is Selected then Condition C_1 is Applied On Attribute A_1.

eg A_1 (Attribute) : Rating

C_1 (Condition) Rating ≥ 9.

LHS

L.H.S

$\sigma_{C_1}\left[\pi_{A_1}(R)\right]$

L.H.S

Rating
9
10

$\sigma_{Rating \geq 9}\left[\pi_{Rating}(\cancel{Re})\right]$

9
8
5
10

1 Reserves (R_1)

Sid	Bid	day
22	101	10/10/96
58	103	11/12/96

2 Sailors(S_1)

Sid	Sname	Rating	age
22	dustin	7	45.0
31	lubber	8	55.5
58	rusty	10	35.0

3 Sailors(S_2)

Sid	Sname	Rating	age
28	yuppy	9	35.0
~~31~~	~~lubber~~	~~8~~	~~55.5~~
~~44~~	~~guppy~~	~~5~~	~~35.0~~
58	rusty	10	35.0

in this eg: L.H.S Can be Replaced / Transformed by R.H.S

(4) $$\boxed{\sigma_{C_1}\left[\underbrace{\pi_{A_1}(R)}_{(L.H.S)}\right] \longrightarrow \underbrace{\pi_{A_1}\left[\sigma_{C_1}(R)\right]}_{(R.H.S)}}$$

YES ✓

In this (L.H.S Parts) firstly Attribute A_1 is Selected then Condition C_1 is Applied on Attribute A_1.

eg A_1 (Attribute) : Rating

C_1 : (Condition) Rating ≥ 9.

RHS

$\pi_{A_1}\left[\sigma_{C_1}(R)\right]$

$\pi_{Rating}\left[\sigma_{Rating \geq 9}(S_2)\right]$

Rating
9
10

		Rating	
28	yuppy	9	35.0
58	rusty	10	35.0

5

$$\boxed{\Pi_A\left(\Pi_{AB}(R)\right) \equiv \Pi_A(R)}$$

6

$$\boxed{\Pi_a(R) \equiv \Pi_a\left(\Pi_b(R)\right)}$$ $\boxed{\text{iff } a \subseteq b}$

eg

$$\Pi_{sid}(\text{Student}) \equiv \Pi_{\underset{'a'}{sid}}\left(\Pi_{\underset{'b'}{sid\ sname}}(\text{Student})\right)$$

7

$$\boxed{\pi_a(R) \neq \pi_b(R)}$$

a : (attribute) Name

b : bid

eg

$$\pi_{name}(S_1) \neq \pi_{bid}(S_1)$$

Set operator

U: Union operator

– : Except or minus

∩ : Intersection operator

- To apply set operations relations must be union compatible.
- R and S relations are union compatible
- If and only if-

(i) Arity of R equal to Arity of S and

(ii) Domain of attributes of R must be same as domain of attributes of s respectively.

Example

Example 1:

$\pi_{Sid\ Sname}(\ldots\ldots\ldots) \cap \pi_{Sid}(\ldots\ldots\ldots)$

{Arity not same so, set operation not allowed}

Example 2:

$\pi_{Sid\ Sname}(\ldots\ldots\ldots) \cap \pi_{Sid\ Marks}(\ldots\ldots\ldots)$

{Arity same but Sname domain is different from marks so, not allowed}

Example

$\pi_{Sid\ Sname}(\ldots\ldots\ldots) \cap \pi_{Stud\ ID,\ Stud\ name}(\ldots\ldots\ldots)$

{Arity and domains are same so, allowed for set operation}

1. Set operation on relation:

R

A
2
2
2
3

S

B
2
2
4

$R \cup S : \{x / x \in R \vee x \in S\} \equiv$

A
2
3
4

$R - S : \{x / x \in R \wedge x \notin S\} \equiv$

A
3

$R \cap S : \{x / x \in R \wedge x \in S\} \equiv$

A
2

Assume Relation R & Relation S consist M & N Tuple Respectively

(1) Range of tuples in $R \cup S = \max(M, N)$ to $M + N$

(2) Range of tuples in $R \cap S = \phi$ to $\min(M, N)$

(3) Range of tuples in $R - S = \phi$ to M

(4) Range of tuples in $S - R = \phi$ to N

Union Operation

- Notation: $r \cup s$
- Defined as :

$$r \cup s = \{t | t \in r \text{ or } t \in s\}$$

- For $r \cup s$ to be valid.
  1. r, s must have the same arity (same number of attributes)
  2. The attribute domains must be compatible (example: 2^{nd} column or r deals with the same type of values as does the 2^{nd} column of s)

Example:

To find all courses taught in the Fall 2009 semester, or in the Spring 2010 semester, or in both.

$$\pi_{course\_id}(\sigma_{semester = \text{"Fall"} \wedge year = 2009} (section)) \cup$$
$$\pi_{course\_id}(\sigma_{semester = \text{"Spring"} \wedge year = 2010} (section))$$

Set Difference Operation

- Notation: $r - s$
- Defined as :

$$r - s = \{t \mid t \in r \text{ and } t \notin s\}$$

- Set differences must be taken between compatible relations.
  - r and s must have the same arity
  - attribute domains of r and s must be compatible

Example:

Example: to find all courses taught in the Fall 2009 semester, but not in the Spring 2010 semester

$$\pi_{course\_id}(\sigma_{semester = \text{“Fall”} \wedge\, year = 2009}\ (section)) -$$
$$\pi_{course\_id}(\sigma_{semester = \text{“Spring”} \wedge\, year = 2010}\ (section))$$

Example:

Sailors (S_1)

Sid	Sname	Rating	age
22	dustin	7	45.0
31	lubber	8	55.5
58	rusty	10	35.0

Sailors (S_2)

Sid	Sname	Rating	age
28	Yuppy	9	35.0
31	Lubber	8	55.5
44	Guppy	5	35.0
58	rusty	10	35.0

(1) Union $S1 \cup S2$

(1) Union

Sid	Sname	Rating	age
22	dustin	7	45.0
31	lubber	8	55.5
58	rusty	10	35.0
44	guppy	5	35.0
28	yuppy	9	35.0

$S1 \cup S2$

(2) Set Difference

Sid	Sname	Rating	age
22	dustin	7	45.0

S1 – S2

(3) Intersection

Sid	Sname	Rating	age
31	lubber	8	55.5
58	rusty	10	35.0

$S1 \cap S2$

Relational Algebra

Basic operators

II. Cross product (×):

- R × S : It result all attributes of R followed by all attributes of S, and each record of R paired with every record of S.
- Degree $(R \times S) = \text{Degree}(R) + \text{Degree}(S)$
- $|(R \times S)| - |R| \times |S|$

NOTE:

- Relation R with n tuples and
- Relation S with 0 tuples then
- number of tuples in $R \times S = 0$ tuples

Join ($\bowtie$)

I. Natural join ($\bowtie$)

$R \bowtie S \equiv \pi_{\text{distinct attributes}}(\sigma_{\text{equality between common attributes of R and S}}(R \times S))$

Example:

- T_1 (ABC) and T_2 (BCDE)

$$\therefore \; T_1 \bowtie T_2 = \pi_{ABCDE}\left(\begin{array}{c}\sigma_{T_1 \cdot B} = T_{2 \cdot B}(T_1 \times T_2) \\ \cap T_1 \cdot C = T_2 \cdot C\end{array}\right)$$

- T_1 (AB) and T_2 (CD)

$$\therefore \; T_1 \bowtie T_2 \equiv T_1 \times T_2 = \pi_{ABCD}(T_1 \times T_2)$$

NOTE:

Natural join equal to cross-product if join condition is empty.

Join ($\bowtie$)

II. Conditional Join ($\bowtie_c$)

- $R \bowtie_c S \equiv \sigma_c (R \times S)$

Join (⋈)

III. Outer Joins:

(a) LEFT OUTER JOIN

$R ⟕ S$: It produces

$(R \bowtie S) \cup$ {Records of R those are failed join condition with remaining attributes null}

(b) RIGHT OUTER JOIN (⟖)

$R ⟖ S$: It produces

$(R \bowtie S) \cup$ {Records of S those are failed join condition with remaining attributes null}

(C) FULL OUTER JOIN (⟗)

$R ⟗ S = (R ⟕ S) \cup (R ⟖ S)$

Natural Join ⋈

R

A	B	C
1	2	4
3	2	6

S

B	C	D
2	4	8
2	7	4

$R \times S =$

R.A	R.B	R.C	S.B	S.C	S.D
1	2	4	2	4	8
1	2	4	2	7	4
3	2	6	2	4	8
3	2	6	2	7	4

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

$R \bowtie S =$

A	B	C	D
1	2	4	8

Left Outer Join [$⟕$]

$(R ⟕ S)$

R

A	B	C
1	2	4
3	2	6

S

B	C	D
2	4	8
2	7	4

$(R \bowtie S) =$

A	B	C	D
1	2	4	8

$R ⟕ S =$

A	B	C	D
1	2	4	8
3	2	6	Null

Right Outer Join [$⟖$]

$R ⟖ S =$

A	B	C	D
1	2	4	8
Null	2	7	4

Full Outer Join [⟗]

Full outer join = Left outer join Union Right outer join

R ⟗ S = R ⟕ S ∪ R ⟖ S

A	B	C	D
1	2	4	8
3	2	6	Null

∪

A	B	C	D
1	2	4	8
Null	2	7	4

R ⟗ S =

A	B	C	D
1	2	4	8
3	2	6	Null
Null	2	7	4

Q. Let R and S be two relations with the following schema

R(P, Q, R1, R2, R3)

S(P, Q, S1, S2)

Where {P, Q} is the key for both schemas. Which of the following queries are equivalent?

I. $\pi_P (R \bowtie S)$

II. $\pi_P(R) \bowtie \pi_P(S)$

III. $\pi_P(\pi_{P,Q} (R) \cap \pi_{P,Q}(S))$

IV. $\pi_P(\pi_{P,Q}(R) - (\pi_{P,Q} (R) - \pi_{P,Q} (S)))$

(A) Only I and II

(B) Only I and III

(C) Only I, II and III

(D) Only I, III and IV

THANK YOU!