



COMPUTER SCIENCE

Database Management System

Query Language

Lecture_8



Vijay Agarwal sir



**TOPICS
TO BE
COVERED**

01

Queries

02

SQL Clauses



SQL clauses

SELECT

FROM

WHERE

GROUP BY

HAVING

Set operatorsSet operators

UNION & UNION ALL

EXCEPT(-) & EXCEPT ALL

INTERSECT & INTERSECT ALL

Comparison operators

<, >, <=, >=, <>

↑
Not
equal.Aggregate operators

COUNT, MIN, MAX

AVG, SUM

Other Set operators

ANY, ALL

IN, NOT IN.

Supplier (Sid Sname Rating)

Parts (Pid Pname Color)

Catalog (Sid Pid Cost)

Sid	Sname	Rating
S1	Abhay	9
S2	mukesh	10
S3	Priya	9
S4	Vinod	10
S5	Mahesh	9

Catalog

Sid	Pid	Cost
S1	P1	6K
S1	P2	7K
S2	P3	8K
S4	P4	9K

'Red' ^ 'Green'

Parts

Pid	Pname	Color
P1	A	Red
P2	B	Green
P3	C	Red
P4	D	Yellow

Q.L W.A.Q to find Sid's who supplied some Red color
Parts  some Green Color Parts ?(i) RA
·(ii) SQL.

Jix RA:

$\text{TT}_{\text{Sid}} \left(\begin{array}{l} \sigma_{C.Pid = P.Pid \wedge (\text{Catalog} \times \text{Parts})} \\ P.\text{Color} = \text{Red} \vee \text{Green} \\ P.\text{Color} = \text{Red} \text{ } \text{OR} \text{ } P.\text{Color} = \text{Green} \end{array} \right)$

or

$\text{Tbsid} \left(\begin{array}{l} \text{C.Pid} = \text{P.Pidn} \\ \text{P.Color} = \text{'Red'} \end{array} \right) \cup \text{Tbsid} \left(\begin{array}{l} \text{C.Pid} = \text{P.Pid} \\ \text{P.Color} = \text{'Green'} \end{array} \right)$

(Q.L) W.A.Q to find Sid's who supplied some Red Color Parts OR some Green Color Parts ?(i) RA
 (ii) SQL.

(i) RA:

$T_{Sid} \left\{ \begin{array}{l} C.Pid = P.Pid \wedge [Catalog \times Parts] \\ P.Color = Red \vee Green \\ P.Color = Red \text{ OR } P.Color = Green \end{array} \right\}$

OR

rename $S(TempL)$, $T_{Sid} \left[\begin{array}{l} C.Pid = P.Pid \wedge [Catalog \times Parts] \\ P.Color = 'Red' \end{array} \right]$ $\overline{TempL \cup TempR}$.

$S(TempR)$, $T_{Sid} \left(\begin{array}{l} C.Pid = P.Pid \wedge [Catalog \times Parts] \\ P.Color = 'Green' \end{array} \right)$

(Q.L) W.A.Q to find Sid's who supplied some Red Color
Parts OR some Green Color Parts ?(i) RA
·(ii) SQL.

(ii) ~~SQL~~

```
SELECT Sid
FROM Catalog C, Parts P
WHERE C.Pid = P.Pid AND
PCColor = 'Red' OR 'GREEN'
```

'UNION'

Q W.A.Q to find Sid's who supplied some Red color
Parts **AND** some Green Color Parts ?(i) RA
·(ii) SQL.

(i) R.A

$\Pi_{\text{Sid}} \left(\begin{array}{l} C.Pid = P.Pid \wedge \\ P.Color = \text{'Red'} \wedge \\ P.Color = \text{'Green'} \end{array} \right)$ Catalog X Parts

(ii) SQL:

SELECT C.Sid
FROM Catalog C, Parts P
WHERE C.Pid = P.Pid AND
P.Color = 'RED' AND P.Color = 'Green'

→ TO Tuple

→ 0 Tuple!

Q.2 W.A.Q to find Sid's who supplied some Red color Parts **AND** some Green Color Parts ?(i) RA
 (ii) SQL.



SQL

```
Select c.Sid
From Catalog c , Parts p
WHERE c.Pid = p.Pid AND p.Color = 'Red'
```

Sid
S1
S2

INTERSECT

```
Select c.Sid
FROM Catalog c , Parts p
WHERE c.Pid = p.Pid AND p.Color = 'Green'
```

Sid
S1

Sid
S1

Sid
S1

Red \wedge
Green

Ans

Q) W.A.Q to find (retrieve) SID of student who secured Highest (1st highest) Marks ?

STUDENT

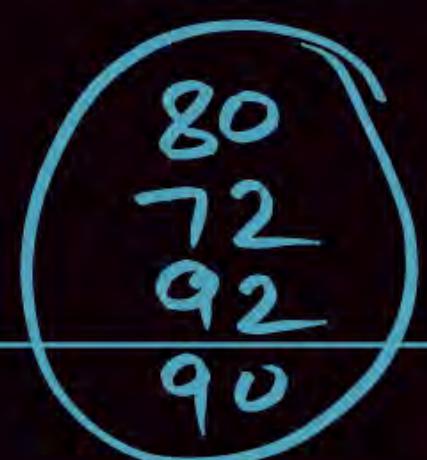
<u>SID</u>	Name	Branch	Marks
S ₁	Khushi	CS	80
S ₂	Denavish	ME	72
S ₃	Rachit	CS	92
S ₄	Bhavya	IT	90

STUDENT (SID, Name, Branch, Marks)

(Sol 1) 1st Highest \Rightarrow S₃ Ans

(Sol 2) 2nd Highest \Rightarrow S₄ Ans

i) R.A



$$T_{Sid}(\text{Student}) = T_{Sid} \left[\text{Student} \bowtie \underset{\text{marks} < M}{\text{marks}} \right]$$

$S_{I,N,B,M}(\text{Student})$

Sid
S ₁
S ₂
S ₃
S ₄

=

Sid
S ₁
S ₂
S ₄

Sid
S ₃

Ano

80	<	80	✓
72	<	72	✓
92	<	92	✓
90	<	90	✓
80	<	80	✓
72	<	72	✓
92	<	92	✓
90	<	90	✓
90	<	80	
72	<	72	
92	<	92	✓
90	<	90	✓

SQL

Sid
S1
S2
S3
S4

SELECT Sid
FROM Student
EXCEPT

SELECT T1.Sid
FROM Student AS T1, Student T2
WHERE T1.marks < T2.Marks

Sid
S1
S2
S4

Q1P

Sid
S3

Ans

S1	80
S2	72
S3	92
S4	90

Using max function

SELECT Sid
FROM Student
WHERE

Marks = $\left(\begin{array}{l} \text{Select max(marks)} \\ \text{FROM Student} \end{array} \right)$

92

Sid
S3

Avg

Q W.A.Q to find (retrieve) Sid of student who secured IInd Highest (2nd highest) Marks ?

STUDENT

<u>SID</u>	Name	Branch	Marks
S ₁	Khushi	CS	80
S ₂	Devansh	ME	72
S ₃	Rachit	CS	92
S ₄	Bhavya	IT	90

STUDENT (SID, Name, Branch, Marks)

Solⁿ²

IInd Highest \Rightarrow S₄ Ans

R.A 2nd Highest Marks

$$S(\underline{\text{Temp}}, \underline{T_{\text{Sid}} \text{ Marks}}) \quad \boxed{\text{Student} \times \text{marks} < m} \quad S_{INBM} (\text{Student})$$

Temp

Sid	Marks
S ₁	80
S ₂	72
S ₄	90

O/P

$T_{\text{Sid}}(\text{Temp}) - T_{\text{Sid}}$ $\boxed{\text{Temp} \times \text{marks} < M} \quad S_{IN} (\text{Temp})$

Temp

Sid
S ₁
S ₂
S ₄

 $-$

Sid
S ₁
S ₂

 \Rightarrow

Sid
S ₄

Avg

Temp

80	72	90
80	72	90 ✓
72	72	90 ✓
90	80 X	90 X

S_1

S_2

~~SQL
2nd Normal~~

Used to SubQuery of p as DB Table

~~WITH~~ Temp (Sid, Marks)

```

SELECT T1.Sid T1.Marks
FROM Student T1, Student T2
WHERE T1.Marks < T2.Marks.

```

Temp	
Sid	Marks
S1	80
S2	72
S4	90

Sid
S1
S2
S4

SELECT Sid
FROM Temp
EXCEPT

Sid
S1
S2

```

SELECT T1.Sid
FROM Temp T1, Temp T2
WHERE T1.Marks < T2.Marks.

```

Sid
S4

90 < 80 80 < 80 90 < 80
X 72 X
S1 S2 S4

90 < 80 80 < 72 90 < 72
X 90 X
S1 S2 S4

72 < 80 ✓ 72 < 90 ✓ 90 < 90 ✓
S2 S4 S4

Using max function (2nd Highest)

```
SELECT Sid  
FROM Student  
WHERE marks =
```

Sid
S4

Avg

(Select max(marks)
FROM Student
WHERE marks <

⑨0

max(marks)

Student

marks <

(Select max(marks)
FROM STUDENT)

⑨2

max(marks)

Q.

Retrieve Sid & Marks of the Student who secured Highest Marks?

~~X~~ **Query I:**

```
Select Sid max(Marks)
From Student
```

Note: Aggregate function can not be in lower clause

→ Syntax Error.

When Aggregate operator & other Attribute (Non Aggregate) used in the Select clause is allowed Only if other Attribute must be in group By clause

~~XX~~ **Query II:**

```
Select Sid, max(marks)
From Student
Group By (Sid).
```

Query III: Select Sid marks
From Student
Where marks =

Sid	Marks
S ₃	90

```
(Select max(Marks))
From Student
```

Syntax Correct but it gives all Sid & their Marks. Not giving Topper Id & Marks.

Query IV

Select Sid. Marks
FROM Student
WHERE

Marks = max(marks)

$S_1 \leftarrow$ O.P.
 $S_2 \leftarrow$ O.P.

$\checkmark 60 = 60$
 $\checkmark 70 = 70$

Table By Table

OTHER SET OPERATOR

1. IN/NOT IN
2. ANY
3. ALL
4. EXISTS/NOT EXISTS

COMPARISION OPERATOR

<, >, <=, <>



Not equal

UNIQUE | NOT UNIQUE

ANY: Compare a value with each value in a Set & Return true if any value is compared according to given condition.

Supplier (Sid, Sname, Rating)

Parts (Pid, Pname, Color)

Catalog (Sid, Pid, Cost)

Supplier (Sid Sname Rating)

Parts (Pid Pname Color)

Catalog (Sid Pid Cost)

Sid	Sname	Rating
S1	Abhay	9
S2	mukesh	10
S3	Priya	9
S4	Vinod	10
S5	Mahesh	9

Catalog

Sid	Pid	Cost
S1	P1	6K
S1	P2	7K
S2	P3	8K
S4	P4	9K

'Red' in Green

Parts

Pid	Pname	Color
P1	A	Red
P2	B	Green
P3	C	Red
P4	D	Yellow.

Q.

Retrieve Sid of the Supplier who supplied some Red Color
Parts?

P
W

✓Query I:

Select
From

Sid
Catalog C, Parts P

output

A diagram showing the output of the query as a table with two rows. The table has a single column and two rows, labeled S_1 and S_2 .

S_1
S_2

WHERE

P.Pid = C.Pid AND

Color = Red

Query II:

Select

From

WHERE

Sid

Catalog

Pid = (

Select Pid
FROM Parts
WHERE Color = Red)

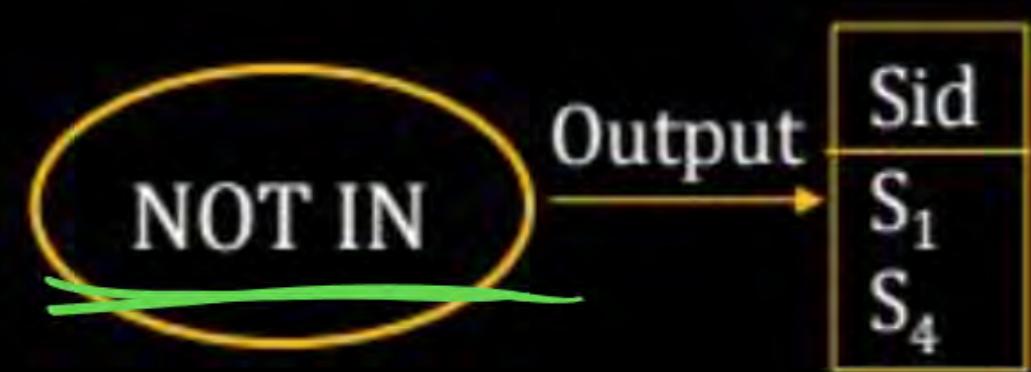
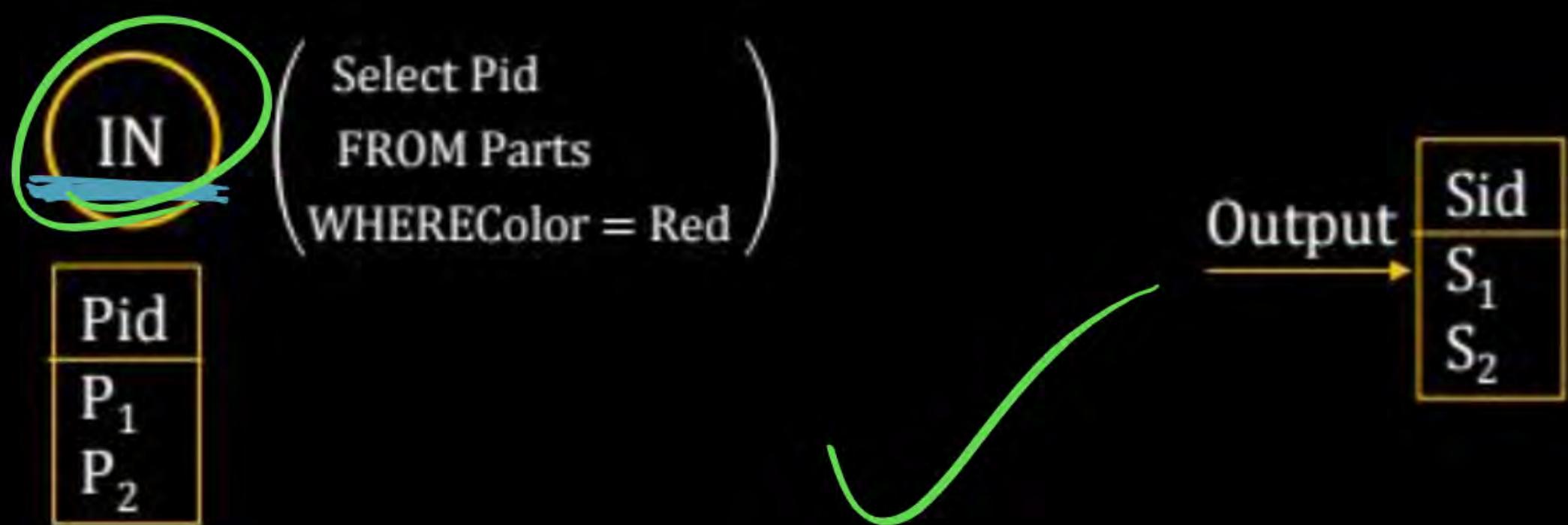
Pid
P ₁
P ₃

One to many
Comparison not
Directly allowed



Query III:

Select Sid
FROM Catalog
WHERE Pid



Nested Query

① Normal (Independent) Nested Query

Execution Sequence

Bottom → Top

Inner Query → Outer Query.

② Co-related Nested Query

Execution Sequence

Top → Bottom → Top.

OUTER QUERY → Inner Query → Outer Query.

Correlated Nested Query Execution.

```
for (i=1 ; i<=n ; i++)
```

```
    for (j=1 ; j<=m ; j++)
```



EXISTS: (Checks): Return True if Inner Query Result Non Empty

EXISTS: (Checks): Return True if Inner Query Result Not Empty

NOT EXIST: Return True if Inner Result Empty

Correlated Nested Query: Inner Query Using attribute defined in Outer Query

Select C.Sid
FROM Catalog C
WHERE EXISTS

(Select *
FROM Part P
WHERE P.Pid =

C.Pid

Inner Query Using the
Attribute Catalog Pid
which is defined in
Outer Query.

Nested Queries

(Independent)
Normal Nest Query

Inner → Outer

Bottom → Top

Corelated Nested
Query

Outer → Inner → Outer

Top → Bottom → Top

```
for(i = 1; i <= n; i++)  
    for(j = 1; j <= m, j++)
```

P
W

$$\begin{array}{c} i = 1 \\ \hline ((((()))) \\ J = 1...m \end{array}$$

$$\begin{array}{c} i = 2 \\ \hline ((((()))) \\ J = 1...m \end{array}$$

$$\begin{array}{c} i = 3 \\ \hline ((((()))) \\ J = 1...m \end{array}$$

.....

$$\begin{array}{c} i = n \\ \hline ((((()))) \\ J = 1...m \end{array}$$

EXIST : Return True if Inner Query Result
Non Empty.

NOT EXIST : Return True if Inner Query Result
Empty.

Corelated Nested Query

Select C.sid
FROM Catalog C

WHERE EXISTS

(Select *
FROM Parts P
WHERE P.Pid = C.Pid
AND Color = Red)

Sid
Red
Color

Inner Query using
Attributes defined in
the Outer Query

OUTER → Inner → Outer
Query

Supplier (Sid Sname Rating)

Parts (Pid Pname Color)

Catalog (Sid Pid Cost)

Sid	Sname	Rating
S1	Abhay	9
S2	mukesh	10
S3	Priya	9
S4	Vinod	10
S5	Mahesh	9

Catalog			
Sid	Pid	Cost	
S1	P1	6K	
S1	P2	7K	
S2	P3	8K	
S4	P4	9K	

Parts			
Pid	Pname	Color	
P1	A	Red	
P2	B	Green	
P3	C	Red	
P4	D	Yellow.	

Corelated Nested Query

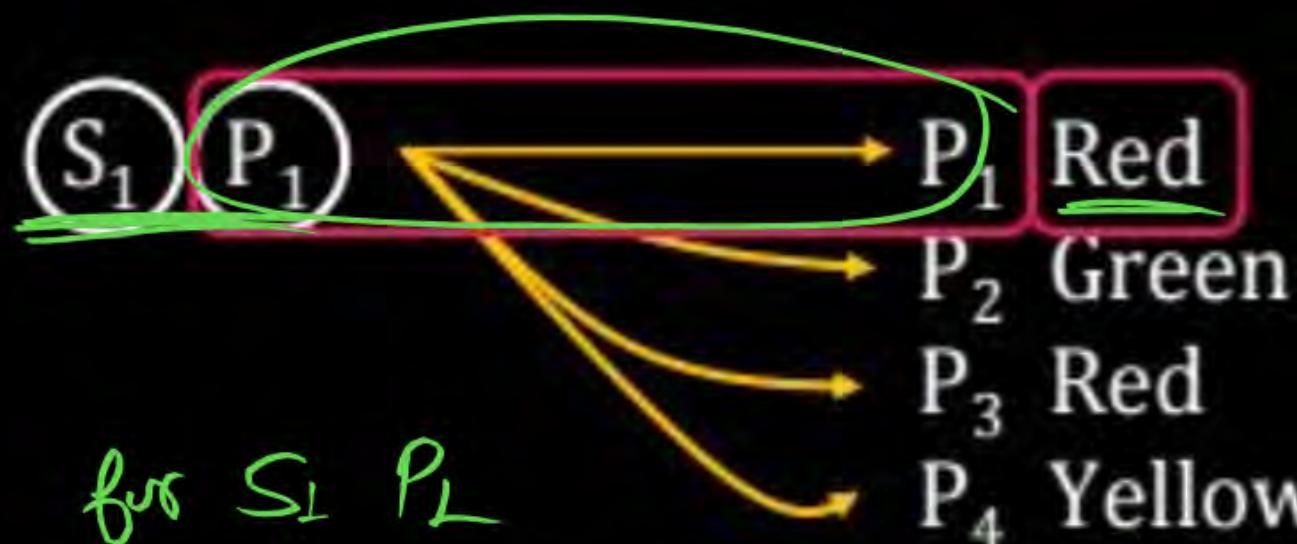
```
Select C.sid  
FROM Catalog C  
WHERE EXISTS (Select *  
               FROM Parts P  
               WHERE P.Pid = C.Pid  
                 AND Color = Red)
```

Sid
Red' Color

Inner Query using
Attributes defined in
the Outer Query

OUTER → Inner → Outer
Query

1st Iteration:



for $S_1 \in P_L$

Pid Match & color = Red

[Pid Match & color = Red]

for $S_1 \neq \text{Non Empty}$

1 Tuple Return

S_1

Catalog

Sid	Pid
S_1	P_1
S_1	P_2
S_2	P_3
S_4	P_4

Parts

Pid	Color
P_1	Red
P_2	Green
P_3	Red
P_4	Yellow

EXIST : True [S1] Select

NOT EXIST : False (Nothing Select)

IInd Iteration:

S_1	P_1	Red
	P_2	<u>Green</u>
	P_3	Red
	P_4	Yellow

[Pid Match but color not Red]

for S_1

0 Tuple Return

Empty

EXIST: False (Nothing Select)

NOT EXIST: True (S_1 Select)

IIIrd Iteration:

S_2	P_1	Red
	P_2	<u>Green</u>
	P_3	Red
	P_4	Yellow

[Pid Match & Color Red]

for S_2

1 Tuple Return

Non Empty

EXIST: True (S_2 Select)

NOT EXIST: False (Nothing Select)

IVth Iteration:

S_4	P_4	P_1 Red
		P_2 Green
		P_3 Red
		P_4 Yellow

Pid Match but color not Red

0 Tuple Return

$\textcircled{S_4}$: Empty

EXIST: False (Nothing Select)

NOTEXIST: True (S_4 Select)

EXIST: Return True if Inner Query Result Non Empty
'Red'

S_1
S_2

NOT EXIST:

S_1
S_4

Color Not 'Red'.

EXISTS

o/p

Sid
S ₁
S ₂

If NOT EXIST then output

Sid
S ₁
S ₄

Before EXIST & NOT EXISTS No Attribute is required.

Before IN & NOT IN Attribute is Required.

Pid IN / Select Pid

Q.

Given Relative Schema

Emp(Eid, Ename, Salary)

Department(Eid, dname, code)

Retrieve Employee ID who have no Department?

Query I: Select Eid

X FROM Emp E, Dep D
WHERE E.Eid <> D.Eid

↑
Not Equal

No Department
OR E_3, E_4 Are

Which is true?

A) $Q_1 \checkmark Q_2 \times$

B) $Q_2 \checkmark Q_1 \times$

C) $Q_1 \checkmark Q_2 \checkmark$

D) $Q_1 \times Q_2 \times$

P
W

Emp. Debt.

Eid	Eid	Dname
E ₁	E ₁	A
E ₂	E ₁	B
E ₃	E ₂	A
E ₄		

~~Query II: Select Eid
FROM Emp E
WHERE NOT EXISTS~~

CROSS Product

Query I:

$E_1 \neq E_1 \rightarrow F$

$E_1 \neq E_1 \rightarrow F$

$E_1 \neq E_2 \rightarrow T$

$E_2 \neq E_1 \rightarrow T$

$E_2 \neq E_1 \rightarrow T$

$E_2 \neq E_2 \rightarrow F$

$E_3 \neq E_1 \rightarrow T$

$E_3 \neq E_1 \rightarrow T$

$E_3 \neq E_2 \rightarrow T$

$E_4 \neq E_1 \rightarrow T$

$E_4 \neq E_1 \rightarrow T$

$E_4 \neq E_2 \rightarrow T$

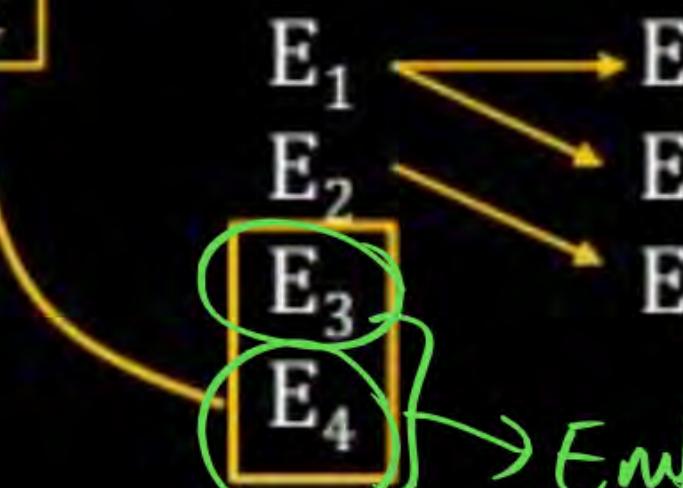
Output of Query I

Eid
E_1
E_2
E_2
E_3
E_3
E_3
E_3
E_4
E_4
E_4

(Select *
FROM Dep D
WHERE E.Eid = D.Eid)

Output of Query II

E_3
E_4



Empty (Inner Query)
NOT EXIST
Return True

NULL:

- Non Zero
- Unknown
- Un existed
- **No Two Null are equal**

Q. Retrieve Eid who does not have any Passport Details?

→ Select Eid
FROM Emp
WHERE PPro = NULL

X

Employee

eid	ename	PassPortNo
E1	A	20
E2	B	NULL
E3	B	16

IS | IS NOT.

NOTE: for comparison with NULL SQL support IS/ IS Not Clause

WHERE PPro ~~IS~~ NULL

Q. Retrieve Eid who are having some passport details?

→ Select Eid
FROM Emp
WHERE PPro **IS NOT NULL**

Regular Expression:

%: Zero or More Character

_ : Exactly One character

Q. Retrieve Student whose Name Start with 'S' & end with M & at least 5 character?

→ 'S _ _ _ % M'

Select *
FROM Student
WHERE Sname = 'S _ _ _ % M' X

S _ _ _ % M
① _ _ _ / _
2 3 4

Output: Pattern → 'S _ _ _ % M' X

Sname LIKE 'S _ _ _ / M'

LIKE: is used to compare to specify certain search Condition for a Pattern in a column.

Select *
FROM Student
WHERE Sname LIKE 'S _ _ _ / M'

Q.

Retrieve all student whose Name NOT Start with 'C'?

P
W

→ Select *
FROM Student
WHERE Sname NOT LIKE 'C%'

C%

```
Select *  
FROM Student  
WHERE Sname
```

LIKE A% → Starts with A

%J → end with J

%I% → Contain I

'____' → All 4 length Name

'S__' → Starts with S & Exactly 4 Length.

exactly 4 length character

'S__%' → Starts with S & at least length is 4. (Minimum 4 length).

- $R \times S$: CROSS Join ✓
- $R \cup S$: Union Join ✓
- $R \bowtie S$: Inner Join ✓
- $R \bowtie S$: Outer Join ✓

ORDER By: Order By Clause is used to sort the Rows.

✓ By Default : Ascending ✓ [ASEC]
Descending [DESC]

Company (Name, invoice no)

Q. Display all the company in alphabetical order of their Name?

Query: Select *
FROM Company
Order by Name asc;

Q. If Reverse alphabetical order then
Order by Name Desc

Q.

P
W

Consider the following relation schemas:

b-Schema = (b-name, b-city, assets)

a-Schema = (a-num, b-name, bal)

d-Schema = (c-name, a-number)

Let branch, account and depositor be respectively instances of the above schemas. Assume that account and depositor relations are much bigger than the branch relation.

Consider the following query:

$$\Pi_{c\text{-name}} (\sigma_{b\text{-city} = \text{"agra"} \wedge \text{bal} < 0} (\text{branch} \bowtie \text{account} \bowtie \text{depositor}))$$



Which one of the following queries is the most efficient version of the above query?

[GATE-2007: 2 Marks]

A

$\Pi_{c\text{-name}} (\sigma_{\text{bal} < 0} (\sigma_{b\text{-city} = \text{"Agra"}} \text{ branch} \bowtie \text{account}) \bowtie \text{depositor})$

B

~~$\Pi_{c\text{-name}} (\sigma_{b\text{-city} = \text{"Agra"}} \text{ branch} \bowtie (\sigma_{\text{bal} < 0} = \text{account}) \bowtie \text{depositor})$~~

C

$\Pi_{c\text{-name}} (\sigma_{b\text{-city} = \text{"Agra"}} \text{ branch} \bowtie \sigma_{b\text{-city} = \text{"Agra} \wedge \text{bal} < 0} \text{ account} \bowtie \text{depositor})$

D

$\Pi_{c\text{-name}} (\sigma_{b\text{-city} = \text{"Agra"}} \text{ branch} \bowtie (\sigma_{b\text{-city} = \text{"Agra} \wedge \text{bal} < 0} \text{ account} \bowtie \text{depositor}))$

Q.

Consider two relations $R_1(A, B)$ with the tuples $(1, 5), (3, 7)$ and
 $R_2(A, C) = (1, 7)(4, 9)$

P
W

Assume that $R(A, B, C)$ is the full natural outer join of R_1 and R_2 .
Consider the following tuples of the form (A, B, C) ; $a = (1, 5, null)$,
 $b = (1, null, 7)$, $c = (3, null, 9)$, $d = (4, 7, null)$, $e = (1, 5, 7)$,
 $f = (3, 7, null)$, $g = (4, null, 9)$. Which one of the following
statements is correct?

[GATE-2015: 1 Mark]

- A R contains a, b, e, f, g, but not c, d
- B R contains all of a, b, c, d, e, f, g
- C R contains e, f, g, but not a, b
- D R contains e but not f, g

Consider a database that has the relation schema CR(StudentName, CourseName). An instance of the schema CR is as given below:

The following query is made on the database.

$T1 \leftarrow \pi_{CourseName}(\sigma_{StudentName='SA'}(CR))$

$T2 \leftarrow CR \div T1;$

The number of rows in $T2$ is _____.

[GATE-2017-CS: 2M]

CR	
Student Name	Course Name
SA	CA
SA	CB
SA	CC
SB	CB
SB	CC
SC	CA
SC	CB
SC	CC
SD	CA
SD	CB

Student Name	Course Name
SD	CC
SD	CD
SE	CD
SE	CA
SE	CB
SF	CA
SF	CB
SF	CC

The following relation records the age of 500 employees of a company, where empNo {indicating the employee number} is the key:

empAge(empNo, age)

Consider the following relational algebra expression:

$\Pi_{empNo}(\text{empAge} \bowtie_{(age > age_1)} \rho_{empNo1, age_1}(\text{empAge}))$

What does the above expression generate? — [GATE-2020-CS: 1M]

emp	age
e1	30
e2	40
e3	50

- A Employee numbers of only those employees whose age is the maximum
- B Employee numbers of only those employees whose age is more than the age of exactly one other employee
- C Employee numbers of all employees whose age is not the minimum
- D Employee numbers of all employees whose age is the minimum

Consider the following relations P(X, Y, Z), Q(X, Y, T) and R(Y, V)

P		
X	Y	Z
X1	Y1	Z1
X1	Y1	Z2
X2	Y2	Z2
X2	Y4	Z4

Q		
X	Y	T
X2	Y1	2
X1	Y2	5
X1	Y1	6
X3	Y3	1

R	
Y	V
Y1	V1
Y3	V2
Y2	V3
Y2	V2

How many tuples will be returned by the following relational algebra query?

$$[\Pi_X(\sigma(P.Y=R.Y \wedge R.V=V2)(P \times R)) - \Pi_X(\sigma(Q.Y=R.Y \wedge Q.T>2)(Q \times R))];$$

[GATE-2019-CS: 2M]

Suppose $R_1(A, B)$ and $R_2(C, D)$ are two relation schemes. Let r_1 and r_2 be the corresponding relation instances. B is a foreign key that refers to C in R_2 . If data in r_1 and r_2 satisfy referential integrity constraints, which of the following is ALWAYS TRUE?

[GATE-2013-CS: 2M]

A $\Pi_B(r_1) \cdot \Pi_C(r_2) = \phi$

B $\Pi_C(r_2) \cdot \Pi_B(r_1) = \phi$

C $\Pi_B(r_1) = \Pi_C(r_2)$

D $\Pi_B(r_1) \cdot \Pi_C(r_2) \neq \phi$

Consider the following table named Student in a relational database. The primary key of this table is rollNum.

Student

Roll Num	Name	Gender	Marks
1	Naman	M	62
2	Aliya	F	70
3	Aliya	F	80
4	James	M	82
5	Swati	F	65

The SQL query below is executed on this database.

```
SELECT *  
FROM Student  
WHERE gender = 'F' AND marks > 65;
```

The number of rows returned by the query is

[GATE-2023-CS: 2M]

Consider the following relation A, B and C:

A		
ID	Name	Age
12	Arun	60
15	Shreya	24
99	Rohit	11

B		
ID	Name	Age
15	Shreya	24
25	Hari	40
98	Rohit	20
99	Rohit	11

C		
ID	Phone	Area
10	2200	02
99	2100	01

How many tuples does the result of the following relational algebra expression contain? Assume that the schema of $A \cup B$ is the same as that of A.

$$(A \cup B) \bowtie_{A.ID > 40 \vee C.ID < 15} C$$

[GATE-2012-CS: 2M]

A 7

B 4

C 5

D 9



Consider the following relation

Cinema (theatre, address, capacity)

Which of the following options will be needed at the end of the SQL query

```
SELECT P1.address  
FROM Cinema P1
```

such that it always finds the addresses of theaters with maximum capacity?

[MCQ: 2015]

- A WHERE P1.capacity >= ALL (select P2.capacity from Cinema P2)
- B WHERE P1.capacity >= Any (select P2.capacity from Cinema P2)
- C WHERE P1.capacity > ALL (select max (P2.capacity) from Cinema P2)
- D WHERE P1.capacity > Any (select max (P2.capacity) from Cinema P2)

Q.

Database table by name Loan_Records is given below:
What is the output of the following SQL query?

HW-4

```
SELECT count(*)
FROM (
    SELECT Borrower, Bank_Manager
    FROM Loan_Records) AS S NATURAL JOIN
    (SELECT Bank_Manager, Loan_Amount
    FROM Loan_Records) AS T
);
```

[MCQ: 2011:2M]

- A 3
- B 9
- C 5
- D 6

Borrower	Bank_Manager	Loan_Amaount
Ramesh	Sunderajan	10000.00
Suresh	Ramgopal	5000.00
Mahesh	Sunderajan	7000.00

P
W



SELECT operation in SQL is equivalent to

[MCQ: 2015-1M]



- A The selection operation in relational algebra.
- B The selection operation in relational algebra, except that SELECT in SQL retains duplicates.
- C The projection operation in relational algebra.
- D The projection operation in relational algebra, except that SELECT in SQL retains duplicates.

Q.

HW-5

Consider the following database table named top_scorer.

Consider the following SQL Query:

```
SELECT ta.player FROM top_scorer AS ta
WHERE ta.goals > ALL (SELECT tb.goals
                      FROM top_scorer AS tb
                      WHERE tb.country = 'Spain')
AND ta.goals > ANY (SELECT tc.goals
                     FROM top_scorer AS tc
                     WHERE tc.country = 'Germany')
```

The number of tuples returned by the above SQL query is ____.

[NAT:2017-2M]

top_scorer		
player	country	goals
Klose	Germany	16
Ronaldo	Brazil	15
G Muller	Germany	14
Fountaine	France	13
Pele	Brazil	12
Klinsmann	Germany	11
Kocsis	Hungary	11
Batistuta	Argentina	10
Cubillas	Peru	10
Lato	Poland	10
Lineker	England	10
T Muller	Germany	10
Rahn	Germany	10

Q.

HW-6.

Consider the following relations:

Student	
Roll_No	Student_Name
1	Raj
2	Rohit
3	Raj

Performance			
Roll_No	Course	Marks	
1	Math	80	
1	English	70	
2	Math	75	
3	English	80	
2	Physics	65	
3	Math	80	

Consider the following SQL query.

SELECT S . Student_Name, Sum(P. Marks)

FROM Student S, Performance P

WHERE S. Roll_No = P.Roll_No

Group BY S. Student_Name

The number of rows that will be returned by the SQL query is ____.

[NAT: 2015-2M]

Q.

Consider the relational database with the following four schemes and their respective instances.

W.W → ?

Student(sNo, sName, dNo) **Dept(dNo, dName)**
Course(cNo, cName, dNo) **Register(sNo, cNo)**

Student		
sNo	sName	dNo
S01	James	D01
S02	Rocky	D01
S03	Jackson	D02
S04	Jane	D01
S05	Milli	D02

Dept	
dNo	dName
D01	CSE
D02	EEE

Course		
cNo	cName	dNo
C11	DS	D01
C12	OS	D01
C21	DE	D02
C22	PT	D02
C23	CV	D03

Register	
sNo	cNo
S01	C11
S01	C12
S02	C11
S03	C21
S03	C22
S03	C23
S04	C11
S04	C12
S05	C11
S05	C21

Question Continues in Next Slide

SQL Query:

```
SELECT * FROM Student AS S WHERE NOT EXIST
  (SELECT cNo FROM Course WHERE dNo = "D01".
  -  

  I) EXCEPT MINUS
II) SELECT cNo FROM Register WHERE sNo = S.sNo)
```

The number of rows returned by the above SQL query is _____.

[NAT: 2022: 2M]

Q.

H.W-8

Consider a database that has the relation schema EMP(EmpId, EmpName, and DeptName). An instance of the schema EMP and a SQL query on it are given below:

```
SELECT AVG(EC.Num)
FROM EC
WHERE(DeptName, Num) IN
    (SELECT DeptName, COUNT(EmpId)AS
     EC(DeptName, Num)
    FROM EMP
    GROUP BY DeptName)
```

The output of executing the SQL query is ____.

[NAT:2017-1M]

EMP		
EmpId	EmpName	DeptName
1	XYA	AA
2	XYB	AA
3	XYC	AA
4	XYD	AA
5	XYE	AB
6	XYF	AB
7	XYG	AB
8	XYH	AC
9	XYI	AC
10	XYJ	AC
11	XYK	AD
12	XYL	AD
13	XYM	AE



Q.

A relational database contains two tables Student and Performance as shown below:

HW → 9

Student	
Roll_no	Student_name
1	Amit
2	Priya
3	Vinit
4	Rohan
5	Smita

Performance		
Roll_no	Student_code	Marks
1	A	86
1	B	95
1	C	90
2	A	89
2	C	92
3	C	80

The primary key of the Student table is Roll_no. For the Performance table, the columns Roll_no. and Subject_code together form the primary key. Consider the SQL query given below:

SELECT S.Student_name, sum(P.Marks) FROM Student S, Performance P
WHERE P.Marks > 84 GROUP BY S.Student_name;

The number of rows returned by the above SQL query is _____.

[NAT: 2019–2M]

Q.

Consider the following database table named water_schemes:
 The number of tuples returned by the following SQL query is:



H.W 10

C with total(name, capacity) as

```
select district_name, sum(capacity)
from water_schemes
group by district_name
```

C with total_avg(capacity) as

```
select avg(capacity)
from total
```

select name

from total, total_avg

where total.capacity ≥ total_avg.capacity

water_schemes		
scheme_no	district_name	capacity
1	Ajmer	20
1	Bikaner	10
2	Bikaner	10
3	Bikaner	20
1	Churu	10
2	Churu	20
1	Dungargarh	10

[NAT:2016-2M]

Q.

K.W

Suppose $R_1(A, B)$ and $R_2(C, D)$ are two relation schemas. Let r_1 and r_2 be the corresponding relation instances. B is a foreign key that refers to C in R_2 . If data in r_1 and r_2 satisfy referential integrity constraints, which of the following is **ALWAYS TRUE**?

- A $\Pi_B(r_1) - \Pi_C(r_2) = \phi$
- B $\Pi_C(r_2) - \Pi_B(r_1) = \phi$
- C $\Pi_B(r_1) = \Pi_C(r_2)$
- D $\Pi_B(r_1) - \Pi_C(r_2) \neq \phi$

[MCQ: 2012–2M]

Q.

W.W12

Consider a relational database containing the following schemas.



Catalogue		
sno	pno	cost
S1	P1	150
S1	P2	50
S1	P3	100
S2	P4	200
S2	P5	250
S3	P1	250
S3	P2	150
S3	P5	300
S3	P4	250

Suppliers		
sno	sname	location
S1	M/s Royal furniture	Delhi
S2	M/s Balaji furniture	Bangalore
S3	M/s Premium furniture	Chennai

Parts		
pno	pname	part_spec
P1	Table	Wood
P2	Chair	Wood
P3	Table	Steel
P4	Almirah	Steel
P5	Almirah	Wood

Question Continues in Next Slide

The primary key of each table is indicated by underlining the constituent field

```
SELECT s.no, s.name  
FROM Suppliers s, Catalogue c  
WHERE s.sno = c.sno AND  
    cost > (SELECT AVG (cost)  
            FROM Catalogue  
            WHERE pno = 'p4'  
            GROUP BY pno);
```

The number of rows returned by the above SQL query is

[MCQ: 2020-2M]

- A 4
- B 5
- C 0
- D 2

Consider a database table T containing two columns X and Y each of type integer. After the creation of the table, one record ($X=1, Y=1$) is inserted in the table.

Let MX and MY denote the respective maximum values of X and Y among all records in the table at any point in time. Using MX and MY, new records are inserted in the table 128 times with X and Y values being $MX+1, 2*MY+1$ respectively. It may be noted that each time after the insertion, values of MX and MY change.

What will be the output of the following SQL query after the steps mentioned above are carried out?

SELECT Y FROM T WHERE X=7;

[GATE-2011-CS: 2M]

A 127

B 255

C 129

D 257

**THANK
YOU!**

