

DBMS LAB ASSIGNMENT 5

NAME - AYAAN KHAN

REG NO - 19BCS013

1. Illustrate logical ANY, ALL and LIKE operator- the queries should be relevant to your respective databases 3 queries for each operator. One query explaining the difference between ANY and ALL

The screenshot shows a SQL IDE window titled "SQLQuery1.sql - lo...EDATOR\ayaan (77)". It contains three SQL queries using the ANY operator. The first query selects all columns from T1_Ward where ward_no is less than any ward_no from T1_Ward where ward_id is less than 10. The second query selects all columns from T1_Doctor where doctor_id is less than any doctor_id from T1_Doctor where doctor_address is 'NZ'. The third query selects all columns from T1_Ward where ward_id is less than any ward_id from T1_Ward where ward_type is 'GEN'. Below the queries, the Results tab shows three tables of data. The first table has 3 rows, the second has 1 row, and the third has 7 rows. A status bar at the bottom indicates "Query executed successfully." and "localhost (15.0 RTM) | PREDATOR\ayaan (77) | Hospital | 00:00:00 | 11 rows".

```
SELECT * FROM T1_Ward
WHERE ward_no < ANY (SELECT ward_no
                     FROM T1_Ward
                     WHERE ward_id < 10
                     );

SELECT * FROM T1_Doctor
WHERE doctor_id < ANY (SELECT doctor_id
                      FROM T1_Doctor
                      WHERE doctor_address = 'NZ'
                      );

SELECT * FROM T1_Ward
WHERE ward_id < ANY (SELECT ward_id
                    FROM T1_Ward
                    WHERE ward_type = 'GEN'
                    );
```

ward_id	ward_no	ward_type	doctor_id	
1	12	ICU	117	
2	5	11	CPU	1002
3	9	11	CPU	1002

doctor_id	doctor_name	doctor_address	doctor_degree	doctor_phone
101	ARJUN	UP	MBBS	123456789

ward_id	ward_no	ward_type	doctor_id	
1	12	ICU	117	
2	21	GEN	1008	
3	5	11	CPU	1002
4	8	21	GEN	1008
5	9	11	CPU	1002
6	10	31	CPU	1001
7	15	31	CPU	1001

Query executed successfully. | localhost (15.0 RTM) | PREDATOR\ayaan (77) | Hospital | 00:00:00 | 11 rows

SQLQuery5.sql - lo...EDATOR\ayaan (51))*

```

SELECT * FROM T1_Ward
WHERE ward_no > ALL (SELECT ward_no
                     FROM T1_Ward
                     WHERE ward_id < 4
                     );
SELECT * FROM T1_Doctor
WHERE doctor_id <> ALL (SELECT doctor_id
                      FROM T1_Doctor
                      WHERE doctor_address = 'UK');
SELECT * FROM T1_Ward
WHERE ward_id <> ALL (SELECT ward_id
                     FROM T1_Ward
                     WHERE ward_type = 'GEN'
                     );

```

100 %

Results Messages

ward_id	ward_no	ward_type	doctor_id
10	31	CPU	1001
15	31	CPU	1001
20	31	CPU	1001

doctor_id	doctor_name	doctor_address	doctor_degree	doctor_phone
101	ARJUN	UP	MBBS	123456789
102	JAMES	NZ	MBBS	123456789
103	RAJ	UP	MBBS	123456789
104	DINESH	UP	MBBS	123456789
105	KRISH	UP	MBBS	123456789
106	AKARSH	UP	MBBS	123456789
107	DIVESH	UP	MBBS	123456789
108	RAJESH	UP	MBBS	123456789

ward_id	ward_no	ward_type	doctor_id
1	12	ICU	117
5	11	CPU	1002
9	11	CPU	1002

Query executed successfully. localhost (15.0 RTM) | PREDATOR\ayaan (51) | Hospital | 00:00:00 | 28 rows

SQLQuery3.sql - lo...EDATOR\ayaan (61))*

```

SELECT doctor_name FROM T1_Doctor
WHERE doctor_name like 'a%';
SELECT doctor_name FROM T1_Doctor
WHERE doctor_name like '%a';
SELECT doctor_name FROM T1_Doctor
WHERE doctor_name like '%ya%';

```

100 %

Results Messages

doctor_name
ABHISHEK
ADITYA
AKARSH
ARJUN

doctor_name
ADITYA
DIVYA
SREYA

doctor_name
ADITYA
DIVYA
SREYA

Query executed successfully. localhost (15.0 RTM) | PREDATOR\ayaan (61) | Hospital | 00:00:00 | 10 rows

DIFFERENCE:

SQLQuery1.sql - lo...EDATOR\ayaan (65)*

```
SELECT * FROM T1_Ward
WHERE ward_no < ANY (SELECT ward_no
                     FROM T1_Ward
                     WHERE ward_id < 9 and ward_id > 5
                     );

SELECT * FROM T1_Ward
WHERE ward_no > ALL (SELECT ward_no
                    FROM T1_Ward
                    WHERE ward_id < 9 and ward_id > 5
                    );
```

100 %

Results Messages

	ward_id	ward_no	ward_type	doctor_id
1	1	12	ICU	117
2	5	11	CPU	1002
3	9	11	CPU	1002

	ward_id	ward_no	ward_type	doctor_id
1	10	31	CPU	1001
2	15	31	CPU	1001
3	20	31	CPU	1001

Query executed successfully. localhost (15.0 RTM) PREDATOR\ayaan (65) Hospital 00:00:00 6 rows

2. One query for each Aggregate function

SQLQuery10.sql - L...EDATOR\ayaan (57)*

```
USE Hospital
SELECT AVG(doctor_id) FROM T1_Doctor;
```

100 %

Results Messages

	(No column name)
1	110

Query executed successfully. localhost (15.0 RTM) PREDATOR\ayaan (57) Hospital 00:00:00 1 rows

SQLQuery13.sql - L_EDATOR\ayaan (57)*

```
USE Hospital
SELECT COUNT(*)
FROM T1_Ward
WHERE ward_no>10;
```

100 %

Results Messages

(No column name)

1	9
---	---

Query executed successfully. localhost (15.0 RTM) PREDATOR\ayaan (57) Hospital 00:00:00 1 rows

SQLQuery14.sql - L_EDATOR\ayaan (57)*

```
USE Hospital
SELECT MAX(doctor_id)
FROM T1_Doctor;
```

100 %

Results Messages

(No column name)

1	120
---	-----

Query executed successfully. localhost (15.0 RTM) PREDATOR\ayaan (57) Hospital 00:00:00 1 rows

```
SQLQuery15.sql - L_EDATOR\ayaan (52)
USE Hospital
SELECT MIN(doctor_id)
FROM T1_Doctor;
```

100 %

Results Messages

(No column name)

1	101
---	-----

Query executed successfully. | localhost (15.0 RTM) | PREDATOR\ayaan (52) | Hospital | 00:00:00 | 1 rows

```
SQLQuery16.sql - L_EDATOR\ayaan (52)
USE Hospital
SELECT SUM(doctor_id)
FROM T1_Doctor;
```

100 %

Results Messages

(No column name)

1	2210
---	------

Query executed successfully. | localhost (15.0 RTM) | PREDATOR\ayaan (52) | Hospital | 00:00:00 | 1 rows

3. Illustrate the usage of order by, group by and having clause (2 queries for each case)

ORDERBY.sql - lo...EDATOR\ayaan (52)*

```

SELECT doctor_name FROM T1_Doctor
WHERE doctor_id<1002 ORDER BY doctor_name ASC

SELECT * FROM T1_Ward
WHERE ward_id<5 ORDER BY doctor_id ASC

SELECT doctor_name FROM T1_Doctor
WHERE doctor_id<1002 ORDER BY doctor_name ASC

SELECT * FROM T1_Ward
WHERE ward_id<5 ORDER BY doctor_id DESC

```

100 %

Results Messages

doctor_name	
1	ABHISHEK
2	ADITYA
3	AKARSH
4	ARJUN
5	BUMRAH
6	DINESH
7	DIVESH
8	DIVYA

ward_id	ward_no	ward_type	doctor_id
1	12	ICU	117
2	21	GEN	1008

doctor_name	
1	ABHISHEK
2	ADITYA
3	AKARSH
4	ARJUN
5	BUMRAH
6	DINESH
7	DIVESH
8	DIVYA

ward_id	ward_no	ward_type	doctor_id
2	21	GEN	1008
1	12	ICU	117

Query executed successfully. localhost (15.0 RTM) | PREDATOR\ayaan (52) | Hospital | 00:00:00 | 44 rows

SQLQuery7.sql - lo...EDATOR\ayaan (51)*

```

SELECT doctor_id FROM T1_Doctor
GROUP BY doctor_id HAVING doctor_id < 111 ;

SELECT ward_type FROM T1_Ward
GROUP BY ward_type HAVING ward_type = 'CPU'

```

100 %

Results Messages

doctor_id	
1	101
2	102
3	103
4	104
5	105
6	106
7	107
8	108
9	109
10	110

ward_type	
1	CPU

Query executed successfully. localhost (15.0 RTM) | PREDATOR\ayaan (51) | Hospital | 00:00:00 | 11 rows

4. Use Aggregate function with group by and having.

SQLQuery23.sql - L..EDATOR\ayaan (52))

```
SELECT AVG(doctor_id) FROM T1_Doctor
GROUP BY doctor_address
HAVING doctor_address='UP';
```

100 %

Results Messages

	(No column name)
1	111

SQLQuery24.sql - L..EDATOR\ayaan (56))

```
SELECT count(patient_id) FROM T1_Patient
GROUP BY disease
HAVING disease='CANCER';
```

100 %

Results Messages

	(No column name)
1	1

Query executed successfully. localhost (15.0 RTM) PREDATOR\ayaan (56) Hospital 00:00:00 1 rows

SQLQuery25.sql - L:\EDATOR\ayaan (52)*

```
SELECT max(patient_id) FROM T1_Patient
GROUP BY disease
HAVING disease='CANCER';
```

100 %

Results Messages

(No column name)

1	1001
---	------

Query executed successfully. localhost (15.0 RTM) PREDATOR\ayaan (52) Hospital 00:00:00 1 rows

SQLQuery26.sql - L:\EDATOR\ayaan (65)*

```
SELECT min(doctor_id) FROM T1_Doctor
GROUP BY doctor_degree
HAVING doctor_degree='MBBS';
```

100 %

Results Messages

(No column name)

1	101
---	-----

Query executed successfully. localhost (15.0 RTM) PREDATOR\ayaan (65) Hospital 00:00:00 1 rows

SQLQuery27.sql - L..EDATOR\ayaan (52)*

```

SELECT sum(doctor_id) FROM T1_Doctor
GROUP BY doctor_address
having doctor_address='UP';

```

100 %

Results Messages

(No column name)

1	1776
---	------

Query executed successfully. localhost (15.0 RTM) PREDATOR\ayaan (52) Hospital 00:00:00 1 rows

5. Write at least 3 nested queries using order by, group by and having clause.

SQLQuery10.sql - L..EDATOR\ayaan (52)*

```

SELECT doctor_name,doctor_id FROM T1_Doctor
GROUP BY doctor_name,doctor_id HAVING doctor_id<105 ORDER BY doctor_name ASC

SELECT ward_id,ward_type,doctor_id FROM T1_Ward
GROUP BY ward_id,ward_type,doctor_id HAVING ward_type='GEN' ORDER BY doctor_id DESC

SELECT doctor_name,doctor_address FROM T1_Doctor
GROUP BY doctor_name,doctor_address HAVING doctor_address='NZ' ORDER BY doctor_name DESC

```

100 %

Results Messages

	doctor_name	doctor_id
1	ARJUN	101
2	DINESH	104
3	JAMES	102
4	RAJ	103

	ward_id	ward_type	doctor_id
1	2	GEN	1008
2	8	GEN	1008
3	18	GEN	1008

	doctor_name	doctor_address
1	JAMES	NZ

Query executed successfully. localhost (15.0 RTM) PREDATOR\ayaan (52) Hospital 00:00:00 8 rows

6. Illustrate the Usage of Except, Exists, Not Exists, Union, Intersection.

SQLQuery12.sql - I...EDATOR\ayaan (52) * X

```

SELECT doctor_id FROM T1_Doctor
EXCEPT
SELECT doctor_id FROM T1_Ward

SELECT * FROM T1_Ward
WHERE exists(SELECT doctor_id FROM T1_Doctor WHERE doctor_id <1000 and T1_Ward.doctor_id=T1_Doctor.doctor_id)

SELECT * FROM T1_Ward
WHERE not exists(SELECT doctor_id FROM T1_Doctor WHERE doctor_id <5 and T1_Ward.doctor_id=T1_Doctor.doctor_id)

SELECT doctor_id FROM T1_Ward
UNION
SELECT doctor_id FROM T1_Doctor

SELECT doctor_id FROM T1_Ward
INTERSECT
SELECT doctor_id FROM T1_Doctor

```

100 %

Results Messages

	doctor_id
1	101
2	102
3	103
4	104
5	105
6	106
7	107
8	108

	ward_id	ward_no	ward_type	doctor_id
1	1	12	ICU	117

	ward_id	ward_no	ward_type	doctor_id
1	1	12	ICU	117
2	2	21	GEN	1008
3	5	11	CPU	1002
4	8	21	GEN	1008
5	9	11	CPU	1002
6	10	31	CPU	1001
7	15	31	CPU	1001
8	18	21	GEN	1008

	doctor_id
1	101
2	102
3	103
4	104
5	105
6	106
7	107
8	108

	doctor_id
1	117

Query executed successfully. | localhost (15.0 RTM) | PREDATOR\ayaan (52) | Hospital | 00:00:00 | 53 rows

7. INNER JOIN, LEFT OUTER JOIN, RIGHT OUTER JOIN- 3 queries for each instance.

INNERJOIN.sql - L:\EDATOR\ayaan (52)

```
select T1_Doctor.doctor_id,T1_Doctor.doctor_name
from T1_Doctor
inner join T1_Patient on T1_Doctor.doctor_id=T1_Patient.doctor_id

select T1_Patient.name,T1_Patient.patient_id
from T1_Patient
inner join T1_Doctor on T1_Doctor.doctor_degree='MBBS' AND T1_Doctor.doctor_id=T1_Patient.doctor_id

select T1_Patient.name,T1_Patient.patient_id
from T1_Patient
inner join T1_Doctor on T1_Doctor.doctor_address='UP' AND T1_Doctor.doctor_id=T1_Patient.doctor_id
```

100 %

Results Messages

doctor_id	doctor_name
117	ABHISHEK

name	patient_id
Ravi	1001

name	patient_id
Ravi	1001

Query executed successfully. localhost (15.0 RTM) PREDATOR\ayaan (52) Hospital 00:00:00 3 rows

LEFTOUTER.sql - L:\EDATOR\ayaan (61)

```
select T1_Doctor.doctor_id,T1_Doctor.doctor_name
from T1_Doctor
left outer join T1_Patient on T1_Doctor.doctor_id=T1_Patient.doctor_id

select T1_Patient.name,T1_Patient.patient_id
from T1_Patient
left outer join T1_Doctor on T1_Doctor.doctor_degree='MBBS' AND T1_Doctor.doctor_id=T1_Patient.doctor_id

select T1_Patient.name,T1_Patient.patient_id
from T1_Patient
left outer join T1_Doctor on T1_Doctor.doctor_address='UP' AND T1_Doctor.doctor_id=T1_Patient.doctor_id
```

100 %

Results Messages

doctor_id	doctor_name
101	ARJUN
102	JAMES
103	RAJ
104	DINESH
105	KRISH
106	AKARISH
107	DIVESH
108	RAJESH
109	TRISH

name	patient_id
Ravi	1001

name	patient_id
Ravi	1001

Query executed successfully. localhost (15.0 RTM) PREDATOR\ayaan (61) Hospital 00:00:00 22 rows

SQLQuery18.sql - L..EDATOR\ayaan (52)*

```

select T1_Doctor.doctor_id,T1_Doctor.doctor_name
from T1_Doctor
right outer join T1_Patient on T1_Doctor.doctor_id=T1_Patient.doctor_id

select T1_Patient.name,T1_Patient.patient_id
from T1_Patient
right outer join T1_Doctor on T1_Doctor.doctor_degree='MBBS' AND T1_Doctor.doctor_id=T1_Patient.doctor_id

select T1_Patient.name,T1_Patient.patient_id
from T1_Patient
right outer join T1_Doctor on T1_Doctor.doctor_address='UK' AND T1_Doctor.doctor_id=T1_Patient.doctor_id

```

100 %

Results Messages

doctor_id	doctor_name
117	ABHISHEK

name	patient_id
NULL	NULL
NULL	NULL
NULL	NULL
NULL	NULL
NULL	NULL
NULL	NULL
NULL	NULL
NULL	NULL

name	patient_id
NULL	NULL
NULL	NULL
NULL	NULL
NULL	NULL
NULL	NULL

Query executed successfully. localhost (15.0 RTM) | PREDATOR\ayaan (52) | Hospital | 00:00:00 | 41 rows

8. Use all the above conditions in JOIN as well.

SQLQuery22.sql - L..EDATOR\ayaan (52)*

```

select T1_Doctor.doctor_id,T1_Doctor.doctor_name
from T1_Doctor
join T1_Patient on T1_Doctor.doctor_id=T1_Patient.doctor_id

select T1_Patient.name,T1_Patient.patient_id
from T1_Patient
join T1_Doctor on T1_Doctor.doctor_degree='MBBS' AND T1_Doctor.doctor_id=T1_Patient.doctor_id

select T1_Patient.name,T1_Patient.patient_id
from T1_Patient
join T1_Doctor on T1_Doctor.doctor_address='UP' AND T1_Doctor.doctor_id=T1_Patient.doctor_id

```

100 %

Results Messages

doctor_id	doctor_name
117	ABHISHEK

name	patient_id
Ravi	1001

name	patient_id
Ravi	1001

Query executed successfully. localhost (15.0 RTM) | PREDATOR\ayaan (52) | Hospital | 00:00:00 | 3 rows