

## DBMS LAB ASSIGNMENT

CH. SAMYUKTHA

19BCS030

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 the SQL Server Enterprise Manager interface. The query editor contains the following SQL code:

```
SELECT * FROM StudentDetails WHERE stdid <=ANY(SELECT stdid FROM StudentDetails WHERE stdid<3)
SELECT * FROM StudentDetails WHERE stdname <=ANY(SELECT stdname FROM StudentDetails WHERE stdid<5)
SELECT salary FROM StudentDetails WHERE stdid >=ANY(SELECT stdid FROM StudentDetails WHERE salary>10000)
```

The Results pane displays the output of the query, showing three tables of results:

stdid	stdname	designation	salary	Location
1	suresh	DBMS	25000	chennai
2	rohini	MA	15000	chennai

stdid	stdname	designation	salary	Location
1	suresh	DBMS	25000	chennai
2	rohini	MA	15000	chennai
3	madhava	ETHICS	50000	nagpur
4	mahendra	SE	75000	guntur
5	sateesh	TOC	65000	guntur

salary
25000
15000
50000
75000
65000

The status bar indicates the query was executed successfully.

The screenshot shows the SQL Server Enterprise Manager interface. The query editor contains the following SQL code:

```
SELECT * FROM StudentDetails WHERE stdid <=ALL(SELECT stdid FROM StudentDetails WHERE stdid<3)
SELECT * FROM StudentDetails WHERE stdname <=ALL(SELECT stdname FROM StudentDetails WHERE stdid<5)
SELECT salary FROM StudentDetails WHERE stdid >=ALL(SELECT stdid FROM StudentDetails WHERE salary>10000)
```

The Results pane displays the output of the query, showing three tables of results:

stdid	stdname	designation	salary	Location
1	suresh	DBMS	25000	chennai

stdid	stdname	designation	salary	Location
3	madhava	ETHICS	50000	nagpur

salary
65000

The status bar indicates the query was executed successfully.

SQLQuery3.sql - localhost:master (LAPTOP-HU6N4SCC\SAMYUKTHA CHILAKA (61)) - Microsoft SQL Server Management Studio

Object Explorer

Connect -> localhost (SQL Server 15.0.2000.5 - LAPTOP-HU6N4SCC\SAMYUKTHA CHILAKA (61))

SQLQuery3.sql - L-KTHA CHILAKA (61) \*

```
SELECT * FROM StudentDetails WHERE stdid LIKE 't%s';  
SELECT * FROM StudentDetails WHERE stdname LIKE '%a%';  
SELECT salary FROM StudentDetails WHERE Location LIKE '%chennai';
```

Results

stdid	stdname	designation	salary	Location	
1	3	madhans	ETHICS	50000	raipur
2	4	mahendra	SE	75000	guntur
3	5	sateesh	TOC	65000	guntur

salary

salary	
1	25000
2	15000

Query executed successfully.

localhost (15.0 RTM) LAPTOP-HU6N4SCC\SAMYUK... master 00:00:00 5 rows

2) One query for each Aggregate function.

SQLQuery3.sql - localhost:master (LAPTOP-HU6N4SCC\SAMYUKTHA CHILAKA (61)) - Microsoft SQL Server Management Studio

Object Explorer

Connect -> localhost (SQL Server 15.0.2000.5 - LAPTOP-HU6N4SCC\SAMYUKTHA CHILAKA (61))

SQLQuery3.sql - L-KTHA CHILAKA (61) \*

```
SELECT COUNT(stdid) FROM StudentDetails WHERE stdid = 2;
```

Results

(No column name)	
1	1

Query executed successfully.

localhost (15.0 RTM) LAPTOP-HU6N4SCC\SAMYUK... master 00:00:00 1 rows

SQLQuery3.sql - localhost.master (LAPTOP-HU6N4SCC\SAMYUKTHA CHILAKA (61)) - Microsoft SQL Server Management Studio

File Edit View Query Project Tools Window Help

Object Explorer

- Connect
- localhost (SQL Server 15.0.2000.5 - LAPTOP-HU6N4SCC\SAMYUKTHA CHILAKA (61))
- Databases
- Security
- Server Objects
- Replication
- PolyBase
- Always On High Availability
- Management
- Integration Services Catalogs
- SQL Server Agent (Agent XPs disabled)
- XEvent Profiler

SQLQuery3.sql - L\_KTHA CHILAKA (61) \* SQLQuery2.sql - L\_KTHA CHILAKA (52) \*

```
SELECT MIN(salary) FROM StudentDetails;
```

191 %

Results Messages

	(No column name)
1	15000

Query executed successfully.

localhost (15.0 RTM) LAPTOP-HU6N4SCC\SAMYUK... master 00:00:00 1 rows

Ready

Type here to search

Ln 1 Col 40 Ch 40 INS 83% 03:39 PM 24-03-2021

SQLQuery3.sql - localhost.master (LAPTOP-HU6N4SCC\SAMYUKTHA CHILAKA (61)) - Microsoft SQL Server Management Studio

File Edit View Query Project Tools Window Help

Object Explorer

- Connect
- localhost (SQL Server 15.0.2000.5 - LAPTOP-HU6N4SCC\SAMYUKTHA CHILAKA (61))
- Databases
- Security
- Server Objects
- Replication
- PolyBase
- Always On High Availability
- Management
- Integration Services Catalogs
- SQL Server Agent (Agent XPs disabled)
- XEvent Profiler

SQLQuery3.sql - L\_KTHA CHILAKA (61) \* SQLQuery2.sql - L\_KTHA CHILAKA (52) \*

```
SELECT MAX(salary) FROM StudentDetails;
```

231 %

Results Messages

	(No column name)
1	75000

Query executed successfully.

localhost (15.0 RTM) LAPTOP-HU6N4SCC\SAMYUK... master 00:00:00 1 rows

Ready

Type here to search

Ln 1 Col 11 Ch 11 INS 83% 03:40 PM 24-03-2021

SQLQuery3.sql - localhost.master (LAPTOP-HU6N4SCC\SAMYUKTHA CHILAKA (61)) - Microsoft SQL Server Management Studio

File Edit View Query Project Tools Window Help

Object Explorer

- localhost (SQL Server 15.0.2000.5 - LAPTOP-HU6N4SCC\SAMYUKTHA CHILAKA (61))
- Databases
- Security
- Server Objects
- Replication
- PolyBase
- Always On High Availability
- Management
- Integration Services Catalogs
- SQL Server Agent (Agent XPs disabled)
- XEvent Profiler

SQLQuery3.sql - L\_KTHA CHILAKA (61)\*

```
SELECT SUM(salary) FROM StudentDetails;
```

Results

	(No column name)
1	230000

Messages

Query executed successfully.

localhost (15.0 RTM) LAPTOP-HU6N4SCC\SAMYUK... master 00:00:00 1 rows

Ready

Type here to search

Ln 1 Col 40 Ch 40 INS

82% 03:41 PM 24-03-2021

SQLQuery3.sql - localhost.master (LAPTOP-HU6N4SCC\SAMYUKTHA CHILAKA (61)) - Microsoft SQL Server Management Studio

File Edit View Query Project Tools Window Help

Object Explorer

- localhost (SQL Server 15.0.2000.5 - LAPTOP-HU6N4SCC\SAMYUKTHA CHILAKA (61))
- Databases
- Security
- Server Objects
- Replication
- PolyBase
- Always On High Availability
- Management
- Integration Services Catalogs
- SQL Server Agent (Agent XPs disabled)
- XEvent Profiler

SQLQuery3.sql - L\_KTHA CHILAKA (61)\*

```
SELECT AVG(salary) FROM StudentDetails;
```

Results

	(No column name)
1	46000

Messages

Query executed successfully.

localhost (15.0 RTM) LAPTOP-HU6N4SCC\SAMYUK... master 00:00:00 1 rows

Ready

Type here to search

Ln 1 Col 40 Ch 40 INS

82% 03:42 PM 24-03-2021

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

The screenshot shows the SQL Server Enterprise Manager interface. The left pane displays the Object Explorer with the 'master' database selected. The right pane shows two queries in the SQL Query Editor. The first query is:

```
SELECT salary FROM StudentDetails ORDER BY salary ASC;  
SELECT stdid FROM StudentDetails ORDER BY stdid ASC;
```

The Results pane shows the output of the first query, which is a table with two columns: salary and stdid. The data is as follows:

salary	stdid
15000	1
25000	2
50000	3
65000	4
75000	5

The second query is:

```
SELECT stdname, Sum(salary) FROM StudentDetails GROUP BY stdname;  
SELECT Location, Sum(salary) FROM StudentDetails GROUP BY Location;
```

The Results pane shows the output of the second query, which is a table with two columns: stdname and Location. The data is as follows:

stdname	Sum(salary)
madhavi	50000
malhenda	75000
rohin	15000
sateesh	65000
suresh	25000

The screenshot shows the SQL Server Enterprise Manager interface. The left pane displays the Object Explorer with the 'master' database selected. The right pane shows two queries in the SQL Query Editor. The first query is:

```
SELECT stdname, Sum(salary) FROM StudentDetails GROUP BY stdname;  
SELECT Location, Sum(salary) FROM StudentDetails GROUP BY Location;
```

The Results pane shows the output of the first query, which is a table with two columns: stdname and Location. The data is as follows:

stdname	Sum(salary)
madhavi	50000
malhenda	75000
rohin	15000
sateesh	65000
suresh	25000

The second query is:

```
SELECT stdname, Sum(salary) FROM StudentDetails GROUP BY stdname HAVING Sum(salary) > 50000;  
SELECT Location, Sum(salary) FROM StudentDetails GROUP BY Location HAVING Sum(salary) > 50000;
```

The Results pane shows the output of the second query, which is a table with two columns: stdname and Location. The data is as follows:

stdname	Sum(salary)
malhenda	75000
sateesh	65000

The screenshot shows the Microsoft SQL Server Management Studio interface. The query editor contains the following SQL code:

```
SELECT stdname FROM StudentDetails condition GROUP BY stdname HAVING AVG(salary)>10000
SELECT stdname FROM StudentDetails condition GROUP BY stdname HAVING AVG(salary)<20000
```

The Results pane shows the output of the query:

stdname
madhushree
maheendra
rahini
sateesh
suresh

The status bar at the bottom indicates "Query executed successfully." and "localhost (15.0 RTM) LAPTOP-HU6N4SCC\SAMYUK... master 00:00:00 6 rows".

#### 4) Use Aggregate function with group by and having.

The screenshot shows the Microsoft SQL Server Management Studio interface. The query editor contains the following SQL code:

```
SELECT COUNT(stdid) FROM StudentDetails WHERE stdid= 2 GROUP BY stdname;
SELECT MIN(salary) FROM StudentDetails GROUP BY stdname;
SELECT Max(salary) FROM StudentDetails GROUP BY stdname;
SELECT SUM(salary) FROM StudentDetails GROUP BY stdname;
SELECT AVG(salary) FROM StudentDetails GROUP BY stdname;
```

The Results pane shows the output of the query:

(No column name)
1
1
2
3
4
5
1
2
3
4
5
1
2

The status bar at the bottom indicates "Query executed successfully." and "localhost (15.0 RTM) LAPTOP-HU6N4SCC\SAMYUK... master 00:00:00 1 rows".

SQLQuery3.sql - localhost.master (LAPTOP-HU6N4SCC\SAMYUKTHA CHILAKA (61)) - Microsoft SQL Server Management Studio

File Edit View Query Project Tools Window Help

master

Object Explorer

- localhost (SQL Server 15.0.2000.5 - LAPTOP-HU6N4SCC\SAMYUKTHA CHILAKA (61))
- Databases
- Security
- Server Objects
- Replication
- PolyBase
- Always On High Availability
- Management
- Integration Services Catalogs
- SQL Server Agent (Agent XPs disabled)
- XEvent Profiler

SQLQuery3.sql - L-KTHA CHILAKA (61) \* SQLQuery2.sql - L-KTHA CHILAKA (52) \*

```
SELECT COUNT(stdid) FROM StudentDetails WHERE stdid= 2 GROUP BY stdname;  
SELECT MIN(salary) FROM StudentDetails GROUP BY stdname;  
SELECT Max(salary) FROM StudentDetails GROUP BY stdname;  
SELECT SUM(salary) FROM StudentDetails GROUP BY stdname;  
SELECT AVG(salary) FROM StudentDetails GROUP BY stdname;
```

191 %

Results Messages

1	50000
2	75000
3	15000
4	65000
5	25000

(No column name)

1	50000
2	75000
3	15000
4	65000
5	25000

(No column name)

1	50000
2	75000
3	15000
4	65000
5	25000

Query executed successfully.

localhost (15.0 RTM) LAPTOP-HU6N4SCC\SAMYUK... master 00:00:00 1 rows

Ready

Type here to search

Ln 1 Col 1 INS 65% 04:40 PM 24-03-2021

SQLQuery3.sql - localhost.master (LAPTOP-HU6N4SCC\SAMYUKTHA CHILAKA (61)) - Microsoft SQL Server Management Studio

File Edit View Query Project Tools Window Help

master

Object Explorer

- localhost (SQL Server 15.0.2000.5 - LAPTOP-HU6N4SCC\SAMYUKTHA CHILAKA (61))
- Databases
- Security
- Server Objects
- Replication
- PolyBase
- Always On High Availability
- Management
- Integration Services Catalogs
- SQL Server Agent (Agent XPs disabled)
- XEvent Profiler

SQLQuery3.sql - L-KTHA CHILAKA (61) \* SQLQuery2.sql - L-KTHA CHILAKA (52) \*

```
SELECT COUNT(stdid) FROM StudentDetails condition WHERE stdid= 2 GROUP BY stdname HAVING AVG(salary) >10000;  
SELECT MIN(salary) FROM StudentDetails condition GROUP BY stdname HAVING AVG(salary) >10000;  
SELECT Max(salary) FROM StudentDetails condition GROUP BY stdname HAVING AVG(salary) >10000;  
SELECT SUM(salary) FROM StudentDetails condition GROUP BY stdname HAVING AVG(salary) >10000;  
SELECT AVG(salary) FROM StudentDetails condition GROUP BY stdname HAVING AVG(salary) >10000;
```

143 %

Results Messages

1	1
---	---

(No column name)

1	50000
2	75000
3	15000
4	65000
5	25000

(No column name)

1	50000
2	75000
3	15000
4	65000
5	25000

(No column name)

1	50000
2	75000

Query executed successfully.

localhost (15.0 RTM) LAPTOP-HU6N4SCC\SAMYUK... master 00:00:00 21 rows

Ready

Type here to search

Ln 5 Col 94 Ch 94 INS 62% 04:51 PM 24-03-2021

The screenshot shows the Microsoft SQL Server Management Studio interface. The query editor contains the following SQL code:

```

SELECT COUNT(stdid) FROM StudentDetails condition WHERE stdid= 2 GROUP BY stdname HAVING AVG(salary) >10000;
SELECT MIN(salary) FROM StudentDetails condition GROUP BY stdname HAVING AVG(salary) >10000;
SELECT Max(salary) FROM StudentDetails condition GROUP BY stdname HAVING AVG(salary) >10000;
SELECT SUM(salary) FROM StudentDetails condition GROUP BY stdname HAVING AVG(salary) >10000;
SELECT AVG(salary) FROM StudentDetails condition GROUP BY stdname HAVING AVG(salary) >10000;

```

The Results pane shows the output of the query, which consists of five rows of data:

stdname	Count	Min	Max	Sum	Avg
1	10000	15000	25000	15000	15000
2	75000	15000	25000	15000	15000
3	15000	15000	25000	15000	15000
4	65000	15000	25000	15000	15000
5	25000	15000	25000	15000	15000

The status bar at the bottom indicates "Query executed successfully." and "localhost (15.0 RTM) LAPTOP-HU6N4SCC\SAMYUK... master 00:00:00 21 rows".

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

The screenshot shows the Microsoft SQL Server Management Studio interface. The query editor contains the following SQL code:

```

SELECT stdname, COUNT(*) FROM StudentDetails
WHERE Stdid= ANY(
    SELECT stdid from StudentDetails
    WHERE Location= ANY(
        SELECT Location FROM StudentDetails
        WHERE salary>=20000
    )
)
GROUP BY stdname HAVING stdname LIKE '%a%'
ORDER BY stdname desc;

```

The Results pane shows the output of the query, which consists of three rows of data:

stdname	Count
1	1
2	1
3	1

The status bar at the bottom indicates "Query executed successfully." and "localhost (15.0 RTM) LAPTOP-HU6N4SCC\SAMYUK... master 00:00:00 3 rows".



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

The screenshot shows the SQL Server Enterprise Manager interface. The Object Explorer on the left shows the server structure. The Query Editor window displays the following SQL query:

```
SELECT stdid FROM StudentDetails  
EXCEPT  
SELECT stdid FROM StudentDetails
```

The Results pane shows a single column header 'stdid' with no data rows. The status bar at the bottom indicates 'Query executed successfully.' and 'localhost (15.0 RTM) LAPTOP-HU6N4SCC\SAMYUK... master 00:00:00 0 rows'.

The screenshot shows the SQL Server Enterprise Manager interface. The Query Editor window displays the following SQL query:

```
select stdid, stdName  
from StudentDetails as a  
where exists  
(  
    select * from StudentDetails as b  
    where a.stdid = b.stdid  
    and Location = 'chennai'  
);
```

The Results pane shows a table with two columns: 'stdid' and 'stdName'. The data is as follows:

stdid	stdName
1	suresh
2	rohin

The status bar at the bottom indicates 'Query executed successfully.' and 'localhost (15.0 RTM) LAPTOP-HU6N4SCC\SAMYUK... master 00:00:00 2 rows'.

SQLQuery3.sql - localhost.master (LAPTOP-HU6N4SCC\SAMYUKTHA CHILAKA (61)) - Microsoft SQL Server Management Studio

File Edit View Query Project Tools Window Help

master

Object Explorer

localhost (SQL Server 15.0.2000.5 - LAPTOP-HU6N4SCC\SAMYUKTHA CHILAKA (61))

Databases

Security

Server Objects

Replication

PolyBase

Always On High Availability

Management

Integration Services Catalogs

SQL Server Agent (Agent XPs disabled)

XEvent Profiler

SQLQuery3.sql - L-KTHA CHILAKA (61) \* SQLQuery2.sql - L-KTHA CHILAKA (52) \*

```
select stdid, stdName
from StudentDetails as a
where not exists
(
    select * from StudentDetails as b
    where a.stdid = b.stdid
    and Location = 'mumbai'
);
```

Results

	stdid	stdName
1	1	suresh
2	2	rohini
3	3	madhavai
4	4	mahendra
5	5	sateesh

Messages

Query executed successfully.

localhost (15.0 RTM) LAPTOP-HU6N4SCC\SAMYUK... master 00:00:00 5 rows

Ready

Type here to search

Ln 7 Col 28 Ch 28 INS 52%

05:26 PM 24-03-2021

SQLQuery3.sql - localhost.master (LAPTOP-HU6N4SCC\SAMYUKTHA CHILAKA (61)) - Microsoft SQL Server Management Studio

File Edit View Query Project Tools Window Help

master

Object Explorer

localhost (SQL Server 15.0.2000.5 - LAPTOP-HU6N4SCC\SAMYUKTHA CHILAKA (61))

Databases

Security

Server Objects

Replication

PolyBase

Always On High Availability

Management

Integration Services Catalogs

SQL Server Agent (Agent XPs disabled)

XEvent Profiler

SQLQuery3.sql - L-KTHA CHILAKA (61) \* SQLQuery2.sql - L-KTHA CHILAKA (52) \*

```
SELECT stdid, stdname
FROM StudentDetails
UNION
SELECT salary, stdname
FROM StudentDetails
ORDER BY stdname
```

Results

	stdid	stdname
1	3	madhavai
2	50000	madhavai
3	4	mahendra
4	75000	mahendra
5	2	rohini
6	15000	rohini
7	5	sateesh
8	65000	sateesh
9	1	suresh
10	25000	suresh

Messages

Query executed successfully.

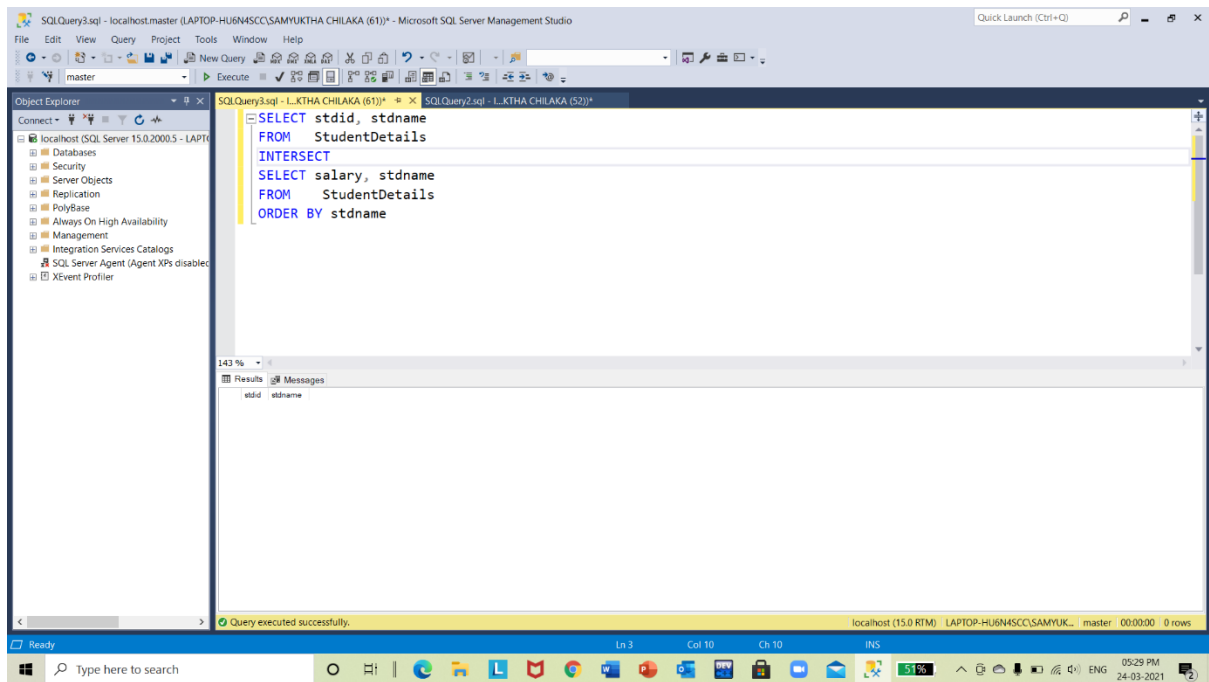
localhost (15.0 RTM) LAPTOP-HU6N4SCC\SAMYUK... master 00:00:00 10 rows

Ready

Type here to search

Ln 6 Col 17 Ch 17 INS 51%

05:29 PM 24-03-2021



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

