

DBMS Lab Assignment 5

Name: Aparna Kholia

Reg No.: 19BCS116

1)

1.1) USE OF ANY

```
SQLQuery1.sql - Io...Aparna Kholia (57)*
SELECT * FROM T1_Room
WHERE Room_num < ANY (SELECT Room_num
                       FROM T1_Room
                       WHERE Patient_ID < 50
                      )
SELECT * FROM T1_Doctor
WHERE doc_ID < ANY (SELECT doc_ID
                   FROM T1_Doctor
                   WHERE doc_specialization = 'ortho'
                  );
SELECT * FROM T1_Patient
WHERE Patient_ID < ANY (SELECT Patient_ID
                       FROM T1_Patient
                       WHERE [Blood Group] = 'O+'
                      );
```

Results

	Room_num	Room_type	Patient_ID
1	101	ICU	11
2	116	General	101
3	204	General	10
4	216	General	100

	doc_ID	doc_name	doc_specialization
1	1	Hani	Ortho
2	2	Mike	Physician
3	3	Ruslan	Child

	Patient_ID	Name	age	Blood Group	Phone_no	app_ID	Room_num
1	1	Akshat	22	B-	257137	564	404
2	10	Vineet	29	O+	555813	301	204
3	11	Grace	51	B+	176796	493	101
4	100	Ankit	23	AB+	971637	927	216

Query executed successfully. localhost (15.0 RTM)

1.2) USE OF ALL

SQLQuery4.sql - Id...Aparna Kholia (52) * X

```
--SELECT * FROM T1_Room
--WHERE Room_num > ALL (SELECT Room_num
--                        FROM T1_Room
--                        WHERE Room_num < 210
--                        );
--SELECT * FROM T1_Doctor
--WHERE doc_ID <> ALL (SELECT doc_ID
--                    FROM T1_Doctor
--                    WHERE doc_name = 'Henri'
--                    );
--SELECT * FROM T1_Room
--WHERE Room_num <> ALL (SELECT Room_num
--                      FROM T1_Room
--                      WHERE Room_type = 'ICU'
--                      );
```

110 %

Results Messages

	Room_num	Room_type	Patient_ID
1	216	General	100
2	404	General	1

	doc_ID	doc_name	doc_specialization
1	4	Divyank	Ortho
2	5	Kush	Phy
3	2	Mike	Physician
4	3	Rustom	Child

	Room_num	Room_type	Patient_ID
1	116	General	101
2	204	General	10
3	216	General	100
4	404	General	1

Query executed successfully. localhost (15.0 RTM)

1.3) USE OF LIKE

SQLQuery6.sql - lo...Apama Kholia (52))

```
--SELECT Name FROM T1_patient
--WHERE Name like 'A%'

--SELECT Name FROM T1_patient
--WHERE Name like '%t'

--SELECT Name FROM T1_patient
--WHERE Name like '%ra%'

|
```

133 %

Results Messages

Name:

1	Akshat
2	Ankit

Name:

1	Akshat
2	Ankit
3	Vinay

Name:

1	Grace
---	-------

Query executed successfully. localhost (15.0 RTM)

DIFFERENCE BETWEEN ANY AND ALL

SQLQuery5.sql - lo...Apama Kholia (61))

```
--SELECT * FROM T1_room
--WHERE Room_num < ANY (SELECT Room_num
--FROM T1_room
--WHERE Room_num < 300 and Room_num > 150
--);

--SELECT * FROM T1_room
--WHERE Room_num > ALL (SELECT Room_num
--FROM T1_room
--WHERE Room_num < 300 and Room_num > 150
--);
```

133 %

Results Messages

Room_num	Room_type	Patient_ID
101	ICU	11
118	General	101
204	General	10

Room_num	Room_type	Patient_ID
404	General	1

Query executed successfully. localhost (15.0 RTM) DESKTOP-MUN98UApama... Hospital

2)

2.1) AVG

The screenshot shows a SQL Query Editor window with the following SQL query:

```
SELECT AVG(doc_ID) FROM T1_appointment;
```

Below the query editor, the Results pane displays the output of the query. The results are as follows:

	(No column name)
1	3

A status bar at the bottom indicates "Query executed successfully." and "localhost (15.0 RTM)".

2.2) COUNT

The screenshot shows a SQL Query Editor window with the following SQL query:

```
SELECT COUNT(*)  
FROM T1_Room  
WHERE Room_num > 110;
```

Below the query editor, the Results pane displays the output of the query. The results are as follows:

	(No column name)
1	8

A status bar at the bottom indicates "Query executed successfully." and "localhost (15.0 RTM) DESKTOP-IMLN98U\Apama..."

2.3) MAX

The screenshot shows a SQL query window with the following text:

```
SELECT MAX(Room_num)
From T1_Patient;
```

Below the query window, the 'Results' tab is active, displaying a single row with the value 404. The status bar at the bottom indicates 'Query executed successfully.' and 'localhost (15.0 RTM) | DESKTOP-JMLN8BU\Apama...'.

	(No column name)
1	404

2.4) MIN

The screenshot shows a SQL query window with the following text:

```
SELECT MIN(Room_num)
From T1_Patient;
```

Below the query window, the 'Results' tab is active, displaying a single row with the value 101. The status bar at the bottom indicates 'Query executed successfully.' and 'localhost (15.0 RTM) | DESKTOP-JMLN8BU\Apama...'.

	(No column name)
1	101

2.5) SUM

The screenshot displays the SQL Server Enterprise Manager interface. The top pane shows a query window with the following SQL code:

```
SELECT SUM(Room_num)
From T1_Patient;
```

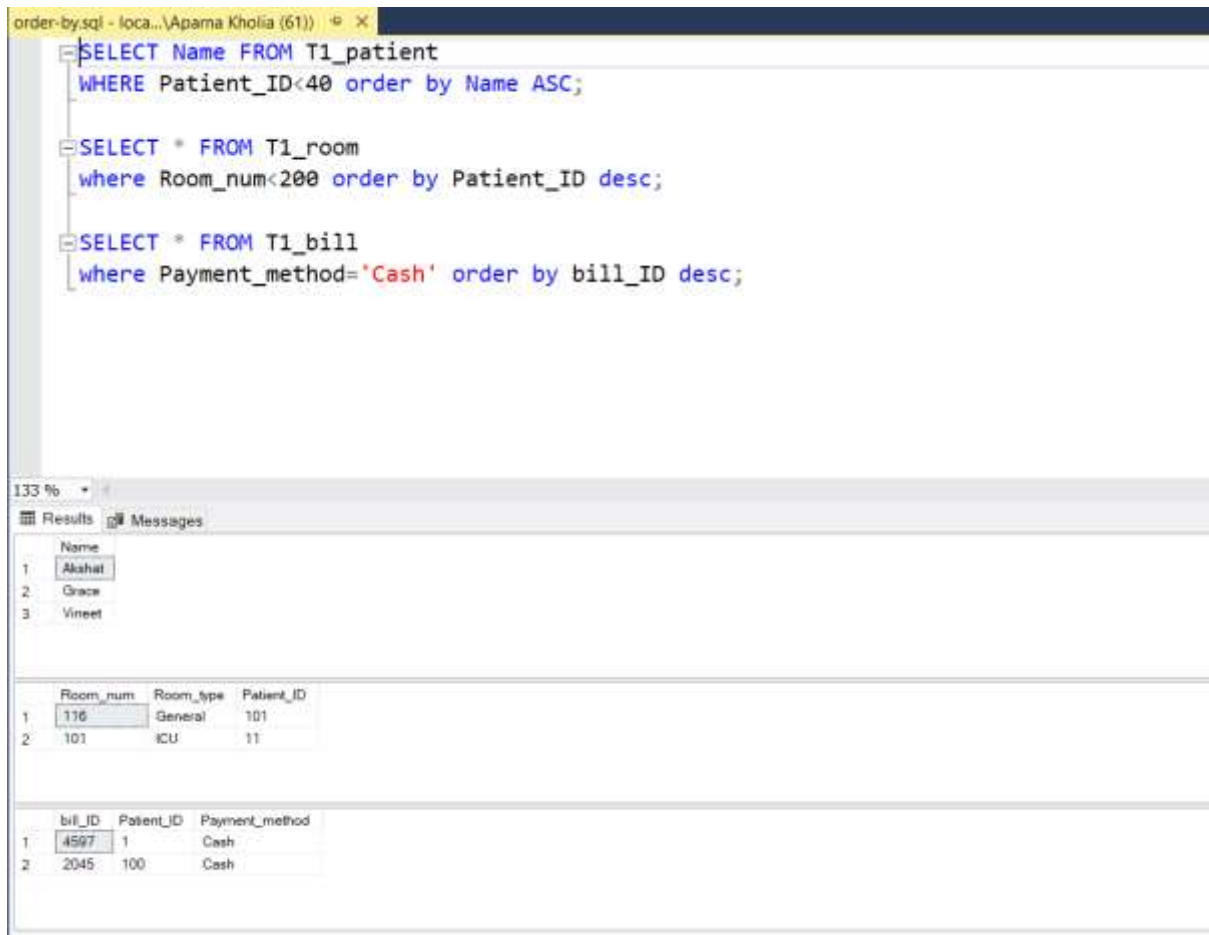
The bottom pane shows the results of the query execution. The results grid contains one row with the value 1041.

1	1041

The status bar at the bottom indicates "Query executed successfully." and "localhost (15.0 RTM) DESKTOP-IMLN988\Apama..."

3)

3.1) ORDER BY



The screenshot shows a SQL IDE window titled 'order-by.sql - loca... \Apama Kholia (61)'. It contains three SQL queries, each with a collapse icon to its left. Below the queries, there are three result sets displayed in a table format. The first result set is for the first query, the second for the second query, and the third for the third query. The IDE interface includes a zoom level of 133% and tabs for 'Results' and 'Messages'.

```
SELECT Name FROM T1_patient
WHERE Patient_ID<40 order by Name ASC;
```

```
SELECT * FROM T1_room
where Room_num<200 order by Patient_ID desc;
```

```
SELECT * FROM T1_bill
where Payment_method='Cash' order by bill_ID desc;
```

Results

	Name
1	Akshat
2	Grace
3	Vineet

	Room_num	Room_type	Patient_ID
1	116	General	101
2	101	ICU	11

	bill_ID	Patient_ID	Payment_method
1	4597	1	Cash
2	2045	100	Cash

3.2) GROUP BY and HAVING

The screenshot shows a SQL query editor with two queries. The first query filters patients by ID, and the second filters payment methods. Below the editor, a table viewer displays the results of these queries.

```
SQLQuery10.sql - I...Aparna Kholia (62))* X
```

```
-- select patient_id from T1_patient  
-- group by patient_id having patient_id < 100 ;  
  
-- select Payment_method from T1_bill  
-- group by Payment_method having Payment_method ='Credit card' ;
```

133 %

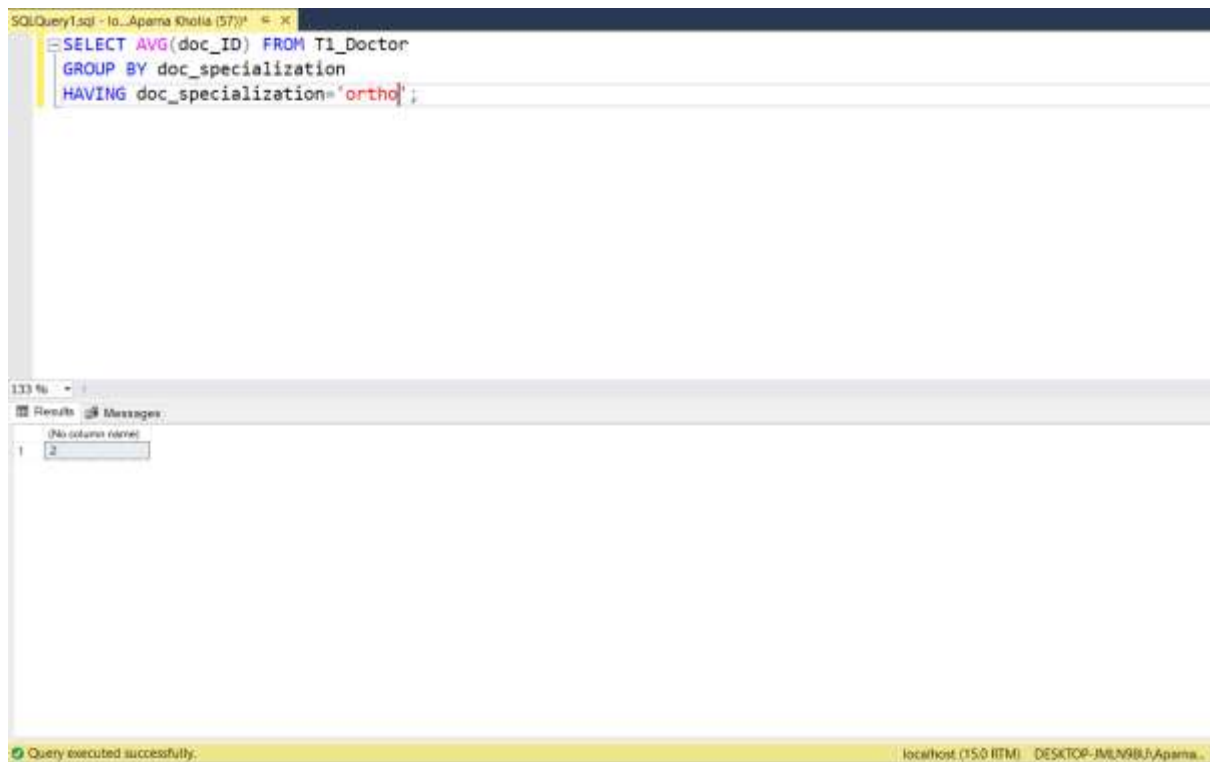
Results Messages

	patient_id
1	1
2	10
3	11

	Payment_method
1	Credit card

4)

4.1) AVG



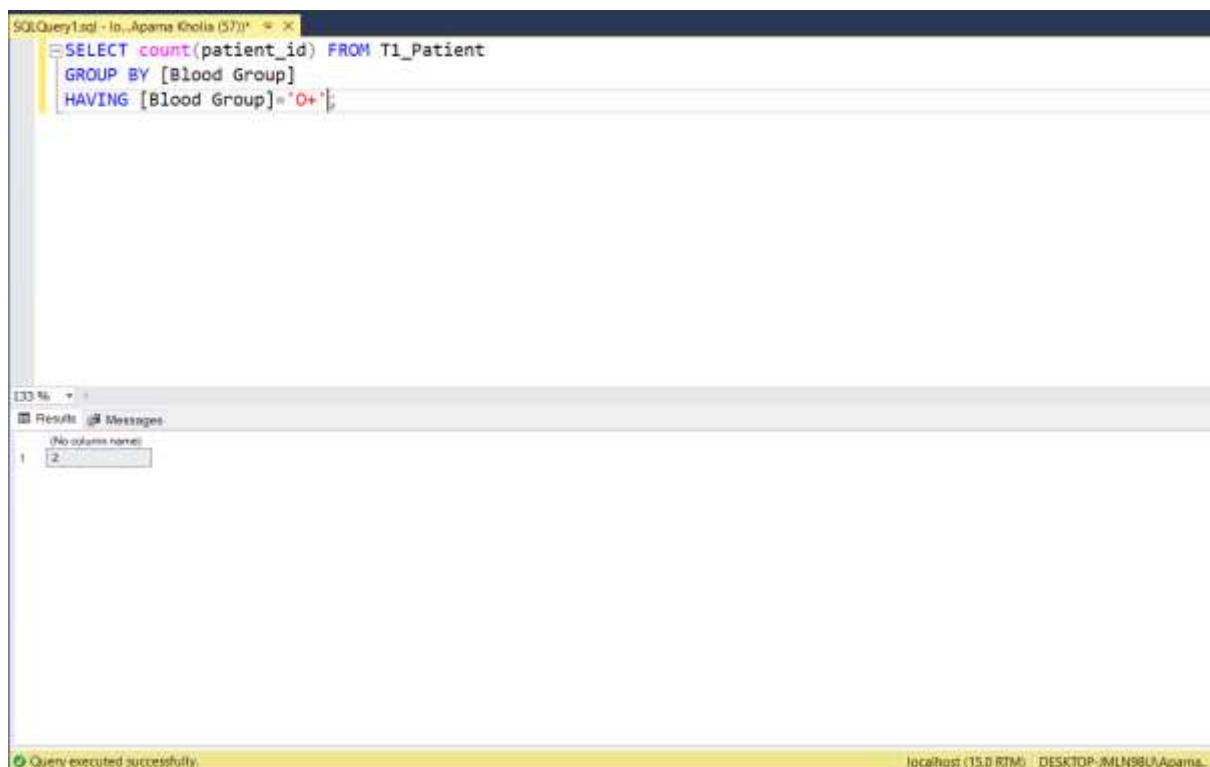
The screenshot shows a SQL query editor window titled "SQLQuery1.sql - lo...Apama Kholia (57)". The query is as follows:

```
SELECT AVG(doc_ID) FROM T1_Doctor  
GROUP BY doc_specialization  
HAVING doc_specialization='ortho';
```

Below the query editor, the "Results" tab is active, showing a single row with the value 2. The status bar at the bottom indicates "Query executed successfully." and the connection details "localhost (15.0 RTM) DESKTOP-JMLN98U\Apama..."

1	2

4.2) COUNT



The screenshot shows a SQL query editor window titled "SQLQuery1.sql - lo...Apama Kholia (57)". The query is as follows:

```
SELECT count(patient_id) FROM T1_Patient  
GROUP BY [Blood Group]  
HAVING [Blood Group]='O+';
```

Below the query editor, the "Results" tab is active, showing a single row with the value 2. The status bar at the bottom indicates "Query executed successfully." and the connection details "localhost (15.0 RTM) DESKTOP-JMLN98U\Apama..."

1	2

4.3) Max

The screenshot shows a SQL query window with the following text:

```
SELECT max(patient_id) FROM T1_Patient  
GROUP BY [Blood Group]  
HAVING [Blood Group]='O+';
```

Below the query window, the Results pane shows a single row with the value 101. The Messages pane is empty.

Query executed successfully. localhost (15.0 RTM) DESKTOP-JMLN9BU\Apama...

4.4) Min

The screenshot shows a SQL query window with the following text:

```
SELECT min(bill_ID) FROM T1_bill  
GROUP BY Payment_method  
HAVING Payment_method='Credit card';
```

Below the query window, the Results pane shows a single row with the value 350. The Messages pane is empty.

Query executed successfully. localhost (15.0 RTM) DESKTOP-JMLN9BU\Apama...

4.5) Sum

SQL Query 12.sql - L_Aparna Khotia (57) * X

```
--SELECT sum(app_ID) FROM T1_appointment  
GROUP BY app_time  
having app_time='12:30';
```

133 %

Results Messages

(No column name)

1	729
---	-----

Query executed successfully. localhost (15.0 RTM) DESKTOP-IML2986/Aparna...

5) Nested queries using ORDER BY, GROUP BY and HAVING

SQL Query 12.sql - L_Aparna Khotia (60) * X

```
--select Name, Patient_ID from T1_Patient  
group by Name, Patient_ID having Patient_ID<50 order by Name desc;  
  
--select Room_num, room_type from T1_room  
group by Room_num, room_type having room_type='general' order by Room_num desc;  
  
--select doc_name, doc_specialization from T1_doctor  
group by doc_name, doc_specialization having doc_specialization='ortho' order by doc_name desc;
```

133 %

Results Messages

	Name	Patient_ID
1	Vinod	10
2	Grace	11
3	Akshat	1

	Room_num	room_type
1	404	General
2	218	General
3	204	General
4	116	General

	doc_name	doc_specialization
1	Hari	Ortho
2	Dhyan	Ortho

Query executed successfully. localhost (15.0 RTM) DES

6)

CODE:

```
SQL Query (sql) - In: Aparna Khola (37)
SELECT doc_ID FROM T1_Doctor
EXCEPT
SELECT doc_ID FROM T1_appointment;

SELECT * FROM T1_appointment
WHERE exists(SELECT doc_ID FROM T1_Doctor WHERE doc_ID <500 and T1_appointment.doc_ID=T1_Doctor.doc_ID) ;

SELECT * FROM T1_appointment
WHERE not exists(SELECT doc_ID FROM T1_Doctor WHERE doc_ID <5 and T1_appointment.doc_ID=T1_Doctor.doc_ID);

SELECT doc_ID FROM T1_appointment
UNION
SELECT doc_ID FROM T1_Doctor;

SELECT doc_ID FROM T1_appointment
INTERSECT
SELECT doc_ID FROM T1_Doctor;
```

OUTPUT:

Results		Messages	
doc_ID			
1	8		
app_ID doc_ID Patient_ID app_time			
1	248	1	101 12:30
2	301	5	10 1:45
3	493	4	11 12:30
4	964	3	1 16:30
5	927	2	100 2:00
app_ID doc_ID Patient_ID app_time			
1	301	5	10 1:45
doc_ID			
1	1		
2	2		
3	3		
4	4		
5	5		
6	6		
doc_ID			
1	1		
2	2		
3	3		
4	4		
5	5		

7)

7.1) Inner join

The screenshot shows three SQL queries executed in SQL Developer, each using an inner join to filter data based on specific criteria.

```
select T1_Doctor.doc_ID,T1_Doctor.doc_name
from T1_Doctor
inner join T1_appointment on T1_Doctor.doc_ID=T1_appointment.doc_ID;

select T1_Patient.name,T1_Patient.patient_id
from T1_Patient
inner join T1_room on T1_room.Room_type='ICU' AND T1_room.Room_num=T1_Patient.Room_num;

select T1_Patient.name,T1_Patient.patient_id
from T1_Patient
inner join T1_appointment on T1_appointment.app_time='12:30' AND T1_appointment.app_ID=T1_Patient.app_ID;
```

The results are displayed in three separate result grids:

doc_ID	doc_name
1	Hari
2	Kushi
3	Duyesh
4	Rudran
5	Mia

name	patient_id
Grace	11

name	patient_id
Grace	11
Elizabeth	101

Query executed successfully. | localhost (15.0 RTM) | DESKTOP-IMN88U\Apama... | Hospital | 000000 | 8 rows

7.2) left outer join

The screenshot shows three SQL queries executed in SQL Developer, each using a left outer join to include all data from the left table and matching data from the right table.

```
select T1_Doctor.doc_ID,T1_Doctor.doc_name
from T1_Doctor
left outer join T1_appointment on T1_Doctor.doc_ID=T1_appointment.doc_ID;

select T1_Patient.name,T1_Patient.patient_id
from T1_Patient
left outer join T1_room on T1_room.Room_type='ICU' AND T1_room.Room_num=T1_Patient.Room_num;

select T1_Patient.name,T1_Patient.patient_id
from T1_Patient
left outer join T1_appointment on T1_appointment.app_time='12:30' AND T1_appointment.app_ID=T1_Patient.app_ID;
```

The results are displayed in three separate result grids:

doc_ID	doc_name
1	Hari
2	Mia
3	Rudran
4	Duyesh
5	Kushi
6	Ravi

name	patient_id
Rudran	1
Anil	100
Elizabeth	101
Grace	11
Shree	10

name	patient_id
Rudran	1
Anil	100
Elia	101
Grace	11
Shree	10

Query executed successfully. | localhost (15.0 RTM) | DESKTOP-IMN88U\Apama... | Hospital | 000000

7.3) Right outer join

```
INNERJOIN.sql - lo..Apama Kholia (60) * X
--select T1_Doctor.doc_ID,T1_Doctor.doc_name
--from T1_Doctor
--right outer join T1_appointment on T1_Doctor.doc_ID=T1_appointment.doc_ID;

--select T1_Patient.name,T1_Patient.patient_id
--from T1_Patient
--right outer join T1_room on T1_room.Room_type='ICU' AND T1_room.Room_num=T1_Patient.Room_num;

--select T1_Patient.name,T1_Patient.patient_id
--from T1_Patient
--right outer join T1_appointment on T1_appointment.app_time='12:30' AND T1_appointment.app_ID=T1_Patient.app_ID;
```

133 % 4

Results Messages

doc_ID	doc_name
1	Henri
5	Kush
4	Diyank
3	Rudran
2	Mike

name	patient_id
Grace	11
NULL	NULL
NULL	NULL
NULL	NULL
NULL	NULL

name	patient_id
Elizabeth	101
NULL	NULL
Grace	11
NULL	NULL
NULL	NULL

Query executed successfully. localhost (15.0 RTM) : DESKTOP-JMLN58U\Apama... Hospital

8) JOIN

```
INNERJOIN.sql - lo..Apama Kholia (60) * X
--select T1_Doctor.doc_ID,T1_Doctor.doc_name
--from T1_Doctor
--join T1_appointment on T1_Doctor.doc_ID=T1_appointment.doc_ID;

--select T1_Patient.name,T1_Patient.patient_id
--from T1_Patient
--join T1_room on T1_room.Room_type='ICU' AND T1_room.Room_num=T1_Patient.Room_num;

--select T1_Patient.name,T1_Patient.patient_id
--from T1_Patient
--join T1_appointment on T1_appointment.app_time='12:30' AND T1_appointment.app_ID=T1_Patient.app_ID;
```

133 % 4

Results Messages

doc_ID	doc_name
1	Henri
5	Kush
4	Diyank
3	Rudran
2	Mike

name	patient_id
Grace	11

name	patient_id
Grace	11
Elizabeth	101

Query executed successfully. localhost (15.0 RTM) : DESKTOP-JMLN58U\Apama... Hospital

