Abhay.N.Rao

19BCS002

DBMS Lab Assignment 5

Q1) 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.

```
a) ANY Operator:-
Select * from Doctor
where DID = ANY(Select Asssigned_Doctor_id from Patient
where Current_Case = 'Kidney Surgery');

Select * from staff
where Assigned_doctor_id = ANY(Select DID from Doctor
where Salary > 30000)
order by Assigned_doctor_id asc;

Select * from Patient
where Asssigned_Doctor_id = ANY(Select Asssigned_Doctor_id from
staff where Assigned_Room between 202 AND 210)
order by Age desc;
```

	DID	First_Name	Last_N	ame	Current_Ca:	se	DPhone_Nu	mber	Qualification	Salary	
1	5	Suresh	Rai		Kidney Surg	gery	11111111114		MBBS	48000	
2	11	Shankar	Mahad	evan	Kidney Sur	gery	11111111120)	MBBS	38000	
	SID	Assigned_d	loctor_id	Name	Salary	Ass	igned_Room				
1	2005	1		Malik	1000	20	8				
2	2001	3		Ayusk	3000	20	0				
3	2011	3		Paran	n 1400	21	5				
4	2008	5		Sujit	2000	213	2				
5	2003	6		Sunn	y 2500	20	4				
6	2002	11		Aak	1100	20:	2				
7	2010	13		Priyar	n 1300	21-	4				
8	2009	15		Anuj	1500	21:	3				
	PID	Patient_nar	ne Age	PPh	one_numbe	r C	Current_Case	Asssig	gned_Doctor_id		
1	1005	Mahesh	36	111	1111134		Brain Surgery	1			
2	1011	Pawan	33	111	1111140		Amnesia	14			
3	1010	Sathwik	32	111	1111139	- 1	Burns	13			
4	1004	Mahendra	32	111	1111133		Skin Surgery	16			
5	1006	Rahul	31	111	1111135		Bone Surgery	9			
6	1001	Abhishek	30	111	1111130	H	Heart Surgery	3			
7	1003	Sundar	29	111	1111132	F	Flu	6			
8	1002	Akash	28	111	1111131	H	Kidney Sur	11			
9	1007	Karthik	27	111	1111136	I	Liver Surgery	12			
10	1008	Surya	26	111	1111137	H	Kidney Sur	5			
11	1009	Maurya	23	111	1111138		Artherities	15			

b) ALL Operator:-

```
Select * from Doctor
where DID > ALL(Select Asssigned_Doctor_id from Patient
where Current_Case = 'Kidney Surgery');

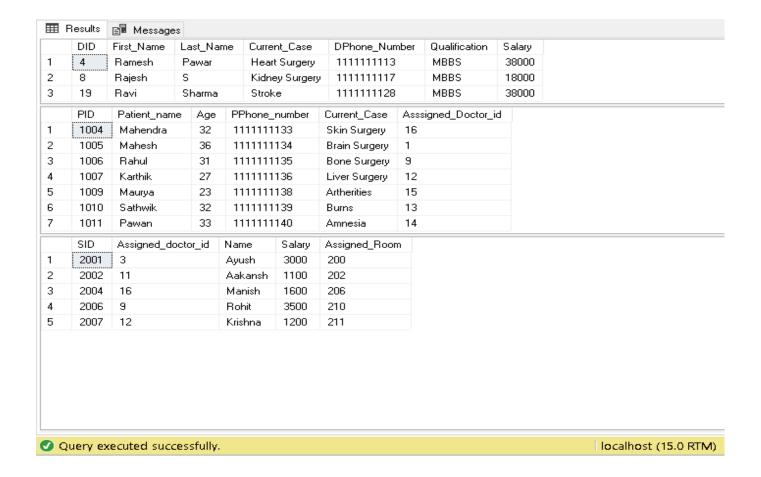
Select * from Patient
where Asssigned_Doctor_id < ALL(Select DID from Doctor
where Salary = 98000)
order by Asssigned_Doctor_id asc;

Select * from Doctor
where Salary > ALL(Select Salary from Doctor
where DID between 9 AND 13)
order by Salary asc;
```

	DID	First_Name	Las	st_Name	Current_Case	DPhone_Numb	oer	Qualification	Salary
1	12	Mithesh	Jai	in	Liver Surgery	11111111121		MBBS	18000
2	13	Sweta	Ku	ımari	Burns	1111111122		MBBS	48000
3	14	Pawar			Amnesia	1111111123		MBBS	98000
4	15	Keshav	Ma	aharaj	Artherities	11111111124		MBBS	68000
5	16	Kishore	Ku	ımar	Skin Surgery	11111111125		MBBS	28000
6	17	Bipul	Ga	autam		11111111126		MBBS	38000
7	18	Ketaki				1111111127		MBBS	38000
8	19	Ravi	Sh	arma	Stroke	1111111128		MBBS	38000
	PID	Patient_nan	ne	Age I	PPhone_number	Current_Case	As	ssigned_Doctor	_id
1	1005	Mahesh		36	1111111134	Brain Surgery	1		
2	1001	Abhishek		30	1111111130	Heart Surgery	3		
3	1008	Surya		26	1111111137	Kidney Surg	5		
4	1003	Sundar		29	1111111132	Flu	6		
5	1006	Rahul		31	1111111135	Bone Surgery	9		
6	1002	Akash		28	1111111131	Kidney Surg	11		
7	1007	Karthik		27	1111111136	Liver Surgery	12	2	
8	1010	Sathwik		32	1111111139	Burns	13	}	
	DID	First_Name	Las	st_Name	Current_Case	DPhone_Numb	oer	Qualification	Salary
1	1	Ajay	Ch	auhan	Brain Surgery	11111111110		MBBS	58000
2	2	Vijay	Sin	ngh	Brain Surgery	11111111111		MBBS	58000
3	15	Keshav	Ma	aharaj	Artherities	11111111124		MBBS	68000
4	7	Mahesh	Pa	rihar	Skin Surgery	11111111116		MBBS	88000
5	14	Pawar			Amnesia	1111111123		MBBS	98000

c) LIKE Operator:-

```
Select * from Doctor where First_Name LIKE 'R%';
Select * from Patient where Patient_name LIKE '_a%';
Select * from staff where Name LIKE '%h%';
```



d) Difference between ANY and ALL:-

```
Select * from Doctor
where Salary > ANY(Select Salary from Doctor
where DID between 9 AND 13);

Select * from Doctor
where Salary > ALL(Select Salary from Doctor
where DID between 9 AND 13);
```

- > ALL () means greater than every value, i.e. greater than the maximum value.
- > ANY() means greater than at least one value, i.e. greater than the minimum value.

Q2) One query for each Aggregate function.

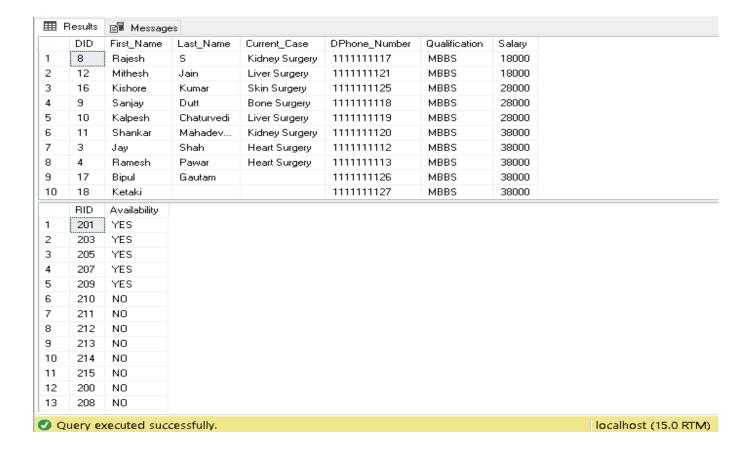
```
Select AVG(Salary) as Avg_Doctor_Salary from Doctor;
Select MAX(Salary) as Max_Staff_Salary from staff;
Select MIN(Salary) as Min_Doctor_Salary from Doctor;
Select COUNT(Availability) as Room_availability from room
where Availability = 'YES';
```



Q3) Illustrate the usage of order by, group by and having clause (2 queries for each case). & Q4) Use Aggregate function with group by and having.

a) Order by case:-

```
Select * from Doctor order by Salary asc;
Select * from room order by Availability desc;
```



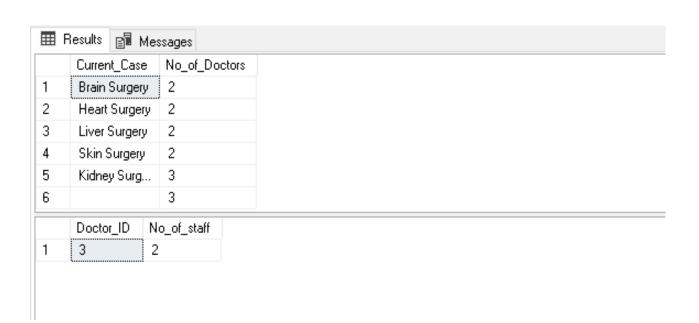
b) Group by case:-

Select Current_Case, COUNT(Current_Case) as No_of_Cases from Patient group by Current_Case order by COUNT(Current_Case) asc; Select COUNT(Availability) as No_of_Rooms, Availability from room group by Availability order by Availability desc;

		No_of_Cases
1	Amnesia	1
2	Artherities	1
3	Bone Surgery	1
4	Brain Surgery	1
5	Burns	1
6	Flu	1
7	Heart Surgery	1
8	Liver Surgery	1
	No_of_Rooms	Availability
1	5	YES
2	11	NO

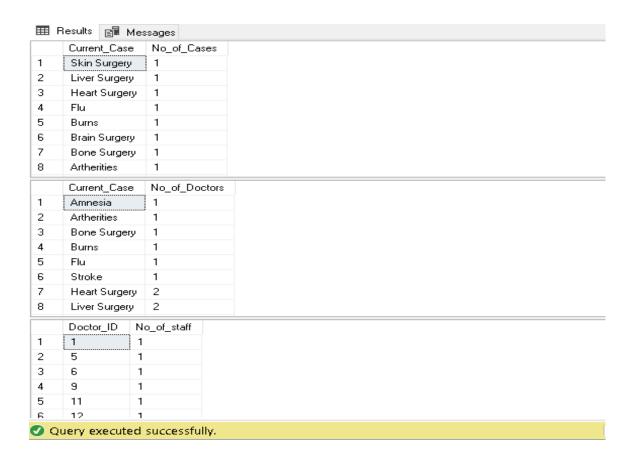
c) Having clause:-

Select Current_Case, COUNT(Current_Case) as No_of_Doctors from
Doctor group by Current_Case having COUNT(Current_Case)>1 order by
COUNT(Current_Case) asc;
Select Assigned_doctor_id as Doctor_ID, COUNT(Assigned_doctor_id)
as No_of_staff from staff group by Assigned_doctor_id having
COUNT(Assigned_doctor_id)>1;



Q5) Write at least 3 nested queries using order by, group by and having clause.

```
Select Current_Case, COUNT(Current_Case) as No_of_Cases from
Patient group by Current_Case having COUNT(Current_Case)<2 order by
Current_Case desc;
Select Current_Case, COUNT(Current_Case) as No_of_Doctors from
Doctor group by Current_Case having COUNT(Current_Case)<3 order by
COUNT(Current_Case) asc;
Select Assigned_doctor_id as Doctor_ID, COUNT(Assigned_doctor_id)
as No_of_staff from staff group by Assigned_doctor_id having
COUNT(Assigned_doctor_id)=1;</pre>
```



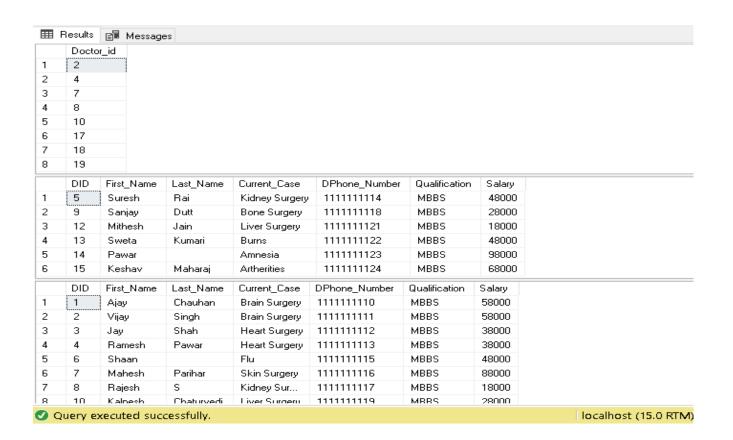
Q6) Illustrate the Usage of Except, Exists, Not Exists, Union, Intersection.

a) Except:-

```
Select DID as Doctor_id from Doctor
EXCEPT
Select Assigned_Doctor_id from Patient;
```

b) Exists:-

```
Select * from Doctor
where EXISTS(Select * from Patient
where PID>1005 and Doctor.DID = Patient.Assigned_Doctor_id);
c) Not Exists:-
Select * from Doctor
where NOT EXISTS(Select * from Patient
where PID>1005 and Doctor.DID = Patient.Assigned_Doctor_id);
d) Union:-
Select CONCAT(First_Name,' ',Last_Name) as Doctor_Name from Doctor
UNION
Select Patient_name from Patient;
e) Intersect:-
Select DID as Doctor_ID from Doctor
INTERSECT
Select Assigned_Doctor_id from Patient;
```



Q7) INNER JOIN, LEFT OUTER JOIN, RIGHT OUTER JOIN- 3 queries for each instance & Q8) Use all the above condition in JOIN as well.

a) INNER JOIN:-

```
Select Doctor.First_Name as Doctor_Name, Patient.Patient_name as
Assigned_Patient_Name
from Doctor
INNER JOIN Patient on Doctor.DID=Patient.Assigned_Doctor_id
order by Doctor_Name asc;

Select Patient.Patient_name, medicine.Med_id as Prescribed_Med_ID
from Patient
INNER JOIN medicine on Patient.PID = medicine.Patient_id
order by Patient_name asc;

Select staff.Name as Staff_Name, Doctor.First_Name as
Assigned_Doctor_Name
from staff
INNER JOIN Doctor on staff.Assigned_doctor_id = Doctor.DID
order by Name asc;
```

l 2	:	Assigned_Patient_Name
2	Ajay	Mahesh
	Jay	Abhishek
3	Keshav	Maurya
4	Kishore	Mahendra
5	Mithesh	Karthik
6	Pawar	Pawan
7	Sanjay	Rahul
8	Shaan	Sundar
	Patient_name	Prescribed_Med_ID
1	Abhishek	100
2	Akash	106
3	Karthik	103
4	Mahendra	107
5	Mahesh	102
6	Maurya	104
7	Pawan	105
8	Rahul	108
	Staff_Name	Assigned_Doctor_Name
1	Aakansh	Shankar
2	Anuj	Keshav
3	Ayush	Jay
4	Krishna	Mithesh
5	Malik	Ajay
6	Manish	Kishore

b) LEFT OUTER JOIN:-

Select Doctor.First_Name as Doctor_Name, Patient.Patient_name as
Assigned_Patient_Name

from Doctor

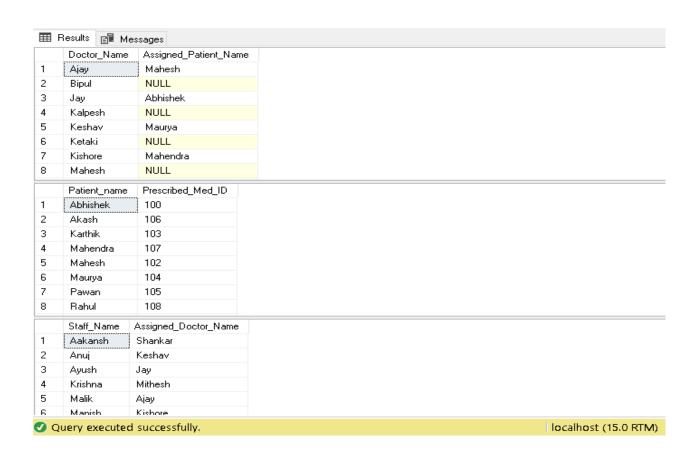
LEFT OUTER JOIN Patient on Doctor.DID=Patient.Assigned_Doctor_id order by Doctor_Name asc;

Select Patient.Patient_name, medicine.Med_id as Prescribed_Med_ID
from Patient

LEFT OUTER JOIN medicine on Patient.PID = medicine.Patient_id
order by Patient_name asc;

Select staff.Name as Staff_Name, Doctor.First_Name as
Assigned_Doctor_Name
from staff

LEFT OUTER JOIN Doctor on staff.Assigned_doctor_id = Doctor.DID order by Name asc;



c) RIGHT OUTER JOIN:-

Select staff.Name as Staff_Name, Doctor.First_Name as
Assigned_Doctor_Name
from staff
RIGHT OUTER JOIN Doctor on staff.Assigned_doctor_id = Doctor.DID
order by Name asc;

Select Doctor.First_Name as Doctor_Name, Patient.Patient_name as
Assigned_Patient_Name

from Doctor

RIGHT OUTER JOIN Patient on Doctor.DID=Patient.Assigned_Doctor_id order by Doctor_Name asc;

Select Patient.Patient_name, medicine.Med_id as Prescribed_Med_ID
from Patient

RIGHT OUTER JOIN medicine on Patient.PID = medicine.Patient_id
order by Patient_name asc;

