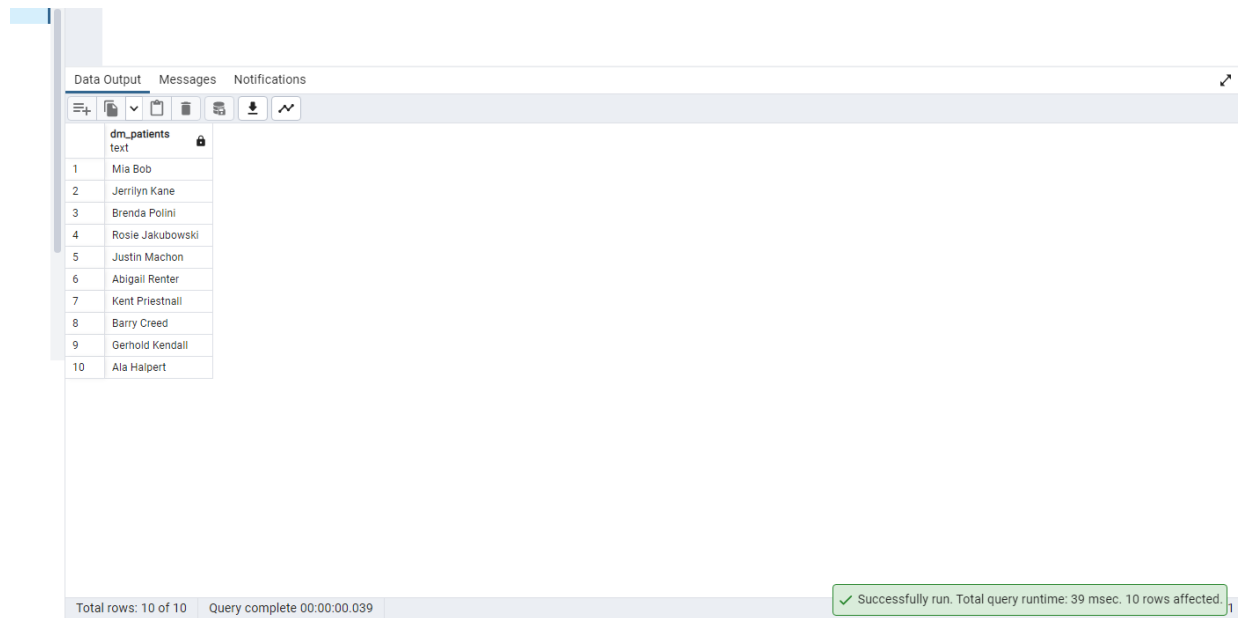


SQL Hackathon Feb 2023- Diabetes DB

1. Display any 10 random DM patients.

Query:

```
select concat("Firstname" || ' ' || "Lastname") DM_Patients from "Patients"
where "Group_ID" = (select "Group_ID" from "Group" where "Group" = 'DM' ) order by random()
limit 10
```



dm_patients
1
Mia Bob
2
Jerrilyn Kane
3
Brenda Polini
4
Rosie Jakubowski
5
Justin Machon
6
Abigail Renter
7
Kent Priestnall
8
Barry Creed
9
Gerhold Kendall
10
Ala Halpert

Total rows: 10 of 10 Query complete 00:00:00.039 ✓ Successfully run. Total query runtime: 39 msec. 10 rows affected.

Note: Since the order by random() function is used in the query, every time the query is executed the result will be different 10 records. So when you run the query to recheck you may not get the above same 10 records

2. Please go through the below screenshot and create the exact output.

Data Output		Explain
	full_name text	
1	Granny Man	
2	Skippie Marriot	
3	Rinaldo Matevosian	
4	Justin Machon	
5	Viv Marcella	

Query:

```
select concat("Firstname" || ' ' || "Lastname") full_name from "Patients" where "Firstname" in ('Granny','Skippie','Rinaldo','Justin','Viv') and "Lastname" like 'M%'
```

Data Output		Messages	Notifications
	full_name text		
1	Granny Man		
2	Skippie Marriot		
3	Rinaldo Matevosian		
4	Justin Machon		
5	Viv Marcella		

Total rows: 5 of 5 Query complete 00:00:00.054 ✓ Successfully run. Total query runtime: 54 msec. 5 rows affected.

3. Write a query to get a list of patients whose RPE start is at moderate intensity.

Query:

```
select concat("Patients"."Firstname" || ' ' || "Patients"."Lastname") Patients_Moderate_RPEStart from "Patients" join "Walking_Test" WT on "Patients"."WalkTest_ID"=WT."WalkTest_ID" where WT."Gait_RPE_Start " between 4 and 6
```

Data Output		Messages	Notifications
patients_moderate_rpestart			
text			
1	Rinaldo Matevosian		
2	Michael Group		
3	Viv Marcella		
Total rows: 3 of 3		Query complete 00:00:00.039	
		✓ Successfully run. Total query runtime: 39 msec. 3 rows affected.	

4. Write a query by using common table expressions and case statements to display birthyear ranges.

Query:

```
with cte as (select date_part ('year',current_date) currentyear)
```

```
select distinct
```

```
case when (cte.currentyear-"Age") between '1940' and '1949' then '1940s'
```

```
when (cte.currentyear-"Age") between '1950' and '1959' then '1950s'
```

```
when (cte.currentyear-"Age") between '1960' and '1969' then '1960s'
```

```
when (cte.currentyear-"Age") between '1970' and '1979' then '1970s'
```

```
else 'other'
```

```
end birthyearranges
```

```
from public."Patients", cte
```

Data Output		Messages	Notifications
birthyearranges			
text			
1	1940s		
2	1950s		
3	1960s		
4	1970s		
Total rows: 4 of 4		Query complete 00:00:00.045	
		✓ Successfully run. Total query runtime: 45 msec. 4 rows affected.	

5. Display DM patient names with highest day MAP and night MAP (without using limit).

--A common method used to estimate the MAP is the following formula: $MAP = DP + \frac{1}{3}(SP - DP)$

Query:

```
select 'HighestDayMap' as MAP,(bp."24Hr_Day_DBP"+((bp."24Hr_Day_SBP"-
bp."24Hr_Day_DBP")/3)) MAP_value,
concat(a."Firstname" || ' ' || a."Lastname") Patient_Name
from "Patients" a
join "Group" g on a."Group_ID"=g."Group_ID"
join "Blood_Pressure" bp on a."BP_ID"=bp."BP_ID"
where g."Group"='DM' and
(bp."24Hr_Day_DBP"+((bp."24Hr_Day_SBP"-bp."24Hr_Day_DBP")/3))=
(select max(("24Hr_Day_DBP"+(("24Hr_Day_SBP"- "24Hr_Day_DBP")/3))) from
public."Blood_Pressure")
```

union all

```
select 'HighestNightMap' as MAP,(bp."24Hr_Night_DBP"+((bp."24Hr_Night_SBP"-
bp."24Hr_Night_DBP")/3)) MAP_value,
concat(a."Firstname" || ' ' || a."Lastname") Patient_Name
from "Patients" a
join "Group" g on a."Group_ID"=g."Group_ID"
join "Blood_Pressure" bp on a."BP_ID"=bp."BP_ID"
where g."Group"='DM' and
(bp."24Hr_Night_DBP"+((bp."24Hr_Night_SBP"-bp."24Hr_Night_DBP")/3))=
(select max(("24Hr_Night_DBP"+(("24Hr_Night_SBP"- "24Hr_Night_DBP")/3))) from
public."Blood_Pressure")
```

64 `select * from "view_question6"`
65

Data Output Messages Notifications

	map	map_value	patient_name
	text	double precision	text
1	HighestDayMap	112.13999938964844	Abigail Renter
2	HighestNightM...	117.52333577473958	Abigail Renter

Total rows: 2 of 2 Query complete 00:00:00.046

✓ Successfully run. Total query runtime: 46 msec. 2 rows affected.

Ln 39, Col 1

6. Create view on table Lab Test by selecting some columns and filter data using Where condition.

Query:

--creating view

create view "view_question6" as select "Patient_ID" Diabetic_Patients,"Fasting_Glucose","Insulin" from "Lab_Test" where "Hb_A1C">6.5 ;

53

Data Output Messages Notifications

CREATE VIEW

Query returned successfully in 56 msec.

Total rows: 27 of 27 Query complete 00:00:00.056

✓ Query returned successfully in 56 msec.

--running the view

select * from "view_question6"

Data Output Messages Notifications				
	diabetic_patients text	Fasting_Glucose integer	Insulin real	
1	S0296	128	4.31	
2	S0301	79	73.96	
3	S0308	112	11.02	
4	S0314	123	1.86	
5	S0318	156	31.88	
6	S0372	118	10.41	
7	S0513	145	44.14	
8	S0522	140	13.07	
9	S0534	122	12.7	
10	S0539	106	17.92	
11	S0540	262	5.77	
12	S0544	147	6.06	
13	S0550	132	5.24	
14	S0552	115	16.17	
15	S0664	124	2.21	
Total rows: 27 of 27		Query complete 00:00:00.051		

✓ Successfully run. Total query runtime: 51 msec. 27 rows affected.

7. Display a list of Patient IDs and their Group whose diabetes duration is greater than 10 years.

Query:

```
select "Patients"."Patient_ID","Group"."Group" from "Patients"
join "Group" on "Patients"."Group_ID"="Group"."Group_ID"
where "Patients"."Diabetes_Duration" >10
```

Data Output Messages Notifications		
	Patient_ID text	Group text
1	S0105	DM
2	S0264	DM
3	S0296	DM
4	S0301	DM
5	S0513	DM
6	S0515	DM
7	S0522	DM
8	S0527	DM
9	S0536	DM
10	S0540	DM
11	S0550	DM
12	S0554	DM
13	S0555	DM
14	S0562	DM
15	S0564	DM
16	S0570	DM
17	S0574	DM
18	S0580	DM
19	S0592	DM
20	S0608	DM
21	S0610	DM
Total rows: 21 of 21		Query complete 00:00:00.038

Ln 55, Col 1

✓ Successfully run. Total query runtime: 38 msec. 21 rows affected.

8. Write a query to list male patient ids and their names who are above 40 years of age and less than 60 years and have Day BloodPressureSystolic above 120 and Day BloodPressureDiastolic above 80.

Query:

```
select "Patients"."Patient_ID",concat("Patients"."Firstname" || ' ' || "Patients"."Lastname")
Male_Patients from "Patients"
```

```

join "Gender" on "Gender"."Gender_ID"="Patients"."Gender_ID"

join "Blood_Pressure" on "Patients"."BP_ID"="Blood_Pressure"."BP_ID"

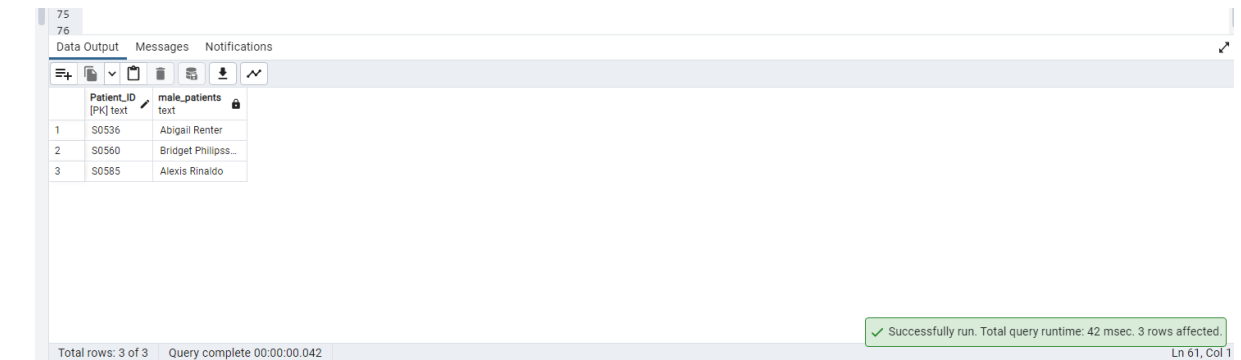
where "Patients"."Age" > 40

and "Patients"."Age"<60 and "Gender"."Gender"='Male'

and "Blood_Pressure"."24Hr_Day_SBP">120

and "Blood_Pressure"."24Hr_Day_DBP">80

```



The screenshot shows a database query execution interface. At the top, there are tabs for 'Data Output', 'Messages', and 'Notifications'. Below the tabs is a toolbar with various icons. The main area displays a table with the following data:

	Patient_ID [PK] text	male_patients text
1	S0536	Abigail Renter
2	S0560	Bridget Phillipss...
3	S0585	Alexis Rinaldo

At the bottom of the interface, there is a status bar that reads: 'Total rows: 3 of 3 Query complete 00:00:00.042'. On the right side, there is a green notification box that says: 'Successfully run. Total query runtime: 42 msec. 3 rows affected. Ln 61, Col 1'.

9. Use a function to calculate the percentage of patients according to the lab visited per month.

Query:

--creating function

create function Calculate_Percentage()

returns table (visitmonth text, Percentage_PatientCount text)

as

\$Body\$

select to_char(LV."Lab_Visit_Date",'Month') visitmonth,

to_char((count("Patient_ID") * 100.0 / sum(count("Patient_ID")) over()),'fm99D00%') from
"Lab_Visit" LV

join "Link_Reference" LR on LV."Lab_visit_ID" = LR."Lab_visit_ID"

join "Patients" on LR."Link_Reference_ID"="Patients"."Link_Reference_ID"

group by visitmonth

\$Body\$

language sql;

```
123 --select date_part('month',"Lab_Visit_Date") visitmonth,
Data Output Messages Notifications
CREATE FUNCTION
Query returned successfully in 64 msec.
```

Total rows: 12 of 12 Query complete 00:00:00.064

✓ Query returned successfully in 64 msec. Ln 107, Col 1

--calling function

select * from Calculate_Percentage()

```
123 --select date_part('month',"Lab_Visit_Date") visitmonth,
Data Output Messages Notifications
```

	visitmonth	percentage_patientcount
1	January	5.19%
2	February	10.39%
3	November	6.49%
4	September	12.99%
5	August	12.99%
6	October	11.69%
7	December	5.19%
8	June	10.39%
9	March	3.90%
10	May	5.19%
11	April	5.19%
12	July	10.39%

Total rows: 12 of 12 Query complete 00:00:00.045

✓ Successfully run. Total query runtime: 45 msec. 12 rows affected. Ln 121, Col 1

10. Count of patients by first letter of firstname.

Query:

select left ("Firstname",1) FirstLetter_Firstname, count("Patient_ID") count_Patients from "Patients" group by FirstLetter_Firstname

```
82 --select nt."Race_ID",count(nt."Patient_ID") from public."Race" R join public."Patients" nt on R."Race_ID"=nt."Race_ID" group by
Data Output Messages Notifications
```

	firstletter_firstname	count_patients
1	B	5
2	J	6
3	Z	1
4	P	3
5	H	3
6	M	5
7	V	1
8	D	7
9	E	1
10	R	4
11	C	4
12	F	3
13	K	4
14	G	7

Total rows: 19 of 19 Query complete 00:00:00.037

✓ Successfully run. Total query runtime: 37 msec. 19 rows affected.

11. write a query to get the list of patients whose lipid test value is null

Query:

```
SELECT CONCAT (a."Firstname" || ' ' || a."Lastname")
FROM "Patients" AS a
RIGHT JOIN "Lipid_Lab_Test" AS b
ON a."Patient_ID"= b."Patient_ID"
WHERE b."Fasting_Cholesterol" is null
OR b."Fasting_Triglyc" is null
OR b."Fasting_HDL" is null
OR b."Fasting_LDL" is null;
```

Data Output	Messages	Notifications
<div> <div>concat</div> <div>text</div> <div>1</div> <div>Rosie Jakubowski</div> <div>2</div> <div>Aridatha Tracy</div> </div>		
<div> <div>✓ Successfully run. Total query runtime: 29 msec. 2 rows affected. ✕</div> </div>		
Total rows: 2 of 2	Query complete 00:00:00.029	Ln 11, Col 28

12. Create a stored procedure to make user ids for the given patient id.

Query:

```
create table Patients_details(Patient_ID varchar(100),User_ID varchar(100));
```

Data Output	Messages	Notifications
<div> <div>CREATE TABLE</div> <div>Query returned successfully in 47 msec.</div> </div>		
Total rows: 30 of 30	Query complete 00:00:00.047	<div> <div>✓ Query returned successfully in 47 msec. ✕</div> </div>

```
select * from public. "Patients_details";
```

Data Output	Messages	Notifications
<div> <div> <div></div> <div></div> <div></div> <div></div> <div></div> <div></div> <div></div> </div> <div> <div></div> <div></div> <div></div> <div></div> <div></div> <div></div> <div></div> </div> </div>		
patient_id character varying (100)	user_id character varying (100)	
<div> <div>Total rows: 0 of 0</div> <div>Query complete 00:00:00.036</div> <div> <div>✓ Successfully run. Total query runtime: 36 msec. 0 rows affected.</div> <div>✕</div> </div> </div>		

CREATE OR REPLACE PROCEDURE Insert_User_ID_details

(

Patient_ID varchar(100),User_ID varchar(100)

)

LANGUAGE plpgsql

AS \$\$

BEGIN

insert into Patients_details values(Patient_ID,User_ID);

END \$\$

Data Output	Messages	Notifications
<div> <div>CREATE PROCEDURE</div> <div>Query returned successfully in 43 msec.</div> </div>		
<div> <div>Total rows: 0 of 0</div> <div>Query complete 00:00:00.043</div> <div> <div>✓ Query returned successfully in 43 msec.</div> <div>✕</div> </div> </div>		

call Insert_User_ID_details('S0030','US0030');

call Insert_User_ID_details('S0033','US0033');

call Insert_User_ID_details('S0064','US0064');

call Insert_User_ID_details('S0068','US0068');

call Insert_User_ID_details('S0078','US0078');

Data Output Messages Notifications

CALL

Query returned successfully in 35 msec.

Total rows: 0 of 0 Query complete 00:00:00.035

✓ Query returned successfully in 35 msec. ✕

select * from public.Patients_details;

Data Output Messages Notifications

Execute/Refresh (FS)

	patient_id character varying (100)	user_id character varying (100)
1	S0030	US0030
2	S0033	US0033
3	S0064	US0064
4	S0068	US0068
5	S0078	US0078
6	S0030	US0030
7	S0033	US0033
8	S0064	US0064
9	S0068	US0068
10	S0078	US0078
11	S0030	US0030
12	S0033	US0033
13	S0064	US0064
14	S0068	US0068
15	S0078	US0078
16	S0030	US0030
17	S0033	US0033

Total rows: 30 of 30 Query complete 00:00:00.037

✓ Successfully run. Total query runtime: 37 msec. 30 rows affected. ✕

13. Show the position of the letter 'r' in the Patient's name.

Query:

```
SELECT concat("Firstname",' ','Lastname") AS patient_name , strpos (concat("Firstname",'
','Lastname'),'r') AS "position_r_letter"

FROM "Patients";
```

Data Output			Messages	Notifications
	patient_name text	position_r_letter integer		
1	Lanni Sue	0		
2	Far Myers	3		
3	Devlin Michael	0		
4	Carmin Texas	3		
5	Tann Bama	0		
6	Jerrilyn Kane	3		
Total rows: 77 of 77		Query complete 00:00:00.041	Ln 27, Col 23	

14. Calculate the patient's birth year in descending order.

Query:

```
Select concat ("Firstname" || ' ' || "Lastname")As "Patient_Fullname",
EXTRACT(YEAR FROM current_Date) - "Age" as "Birth_Year" from "Patients"
order by "Birth_Year" desc;
```

Data Output			Messages	Notifications
	Patient_Fullname text	Birth_Year numeric		
1	Abigail Renter	1973		
2	Diahann Fealty	1973		
3	Joesph Long	1972		
4	Ted Hue	1971		
5	Lolita Darci	1971		
6	Alexis Rinaldo	1971		
Total rows: 77 of 77		Query complete 00:00:00.037	Ln 1	

15.Find the patients that have eye damage due to diabetes.

Query:

```
SELECT CONCAT (b."Firstname" || ' ' || b."Lastname")DM_Patients_with_eye_damage
FROM "Patients" as b
Inner Join "Group"a
On b."Group_ID"= a."Group_ID"
Inner join "Ophthalmology" as c
on b. "Opthal_ID"=c."Opthal_ID"
Where "Group"= 'DM' And "Diabetic_Retinopathy">0 Or "Macular_Edema">0;
```

Data Output		Messages	Notifications
<div> <div>dm_patients_with_eye_damage</div> <div>text</div> </div>			
1	Far Pardon		
2	Gerhold Kendall		
3	Sheena Kevin		
4	Daugherty Conner		
5	Megan Walter		
6	Bailev Kina		
Total rows: 22 of 22		Query complete 00:00:00.041	Ln 47, Col 71

✓ Successfully run. Total query runtime: 41 msec. 22 rows affected. ✕

16. Query to classify Gait RPE End into 5 categories as per the intensity. (Hint: Use of CASE Statement)

Query:

```
select distinct case when (WT."Gait_RPE_End " =0) then 'Rest'
when (WT."Gait_RPE_End " between 1 and 3 ) then 'Easy Intensity'
when (WT."Gait_RPE_End " between 4 and 6 ) then 'Moderate Intensity'
when (WT."Gait_RPE_End " between 7 and 9 ) then 'Hard Intensity'
else 'Effort intensity'
end as "intensity_category"
from "Patients"
join "Walking_Test" WT on "Patients"."WalkTest_ID"=WT."WalkTest_ID"
```

Data Output		Messages	Notifications
<div> <div>intensity_category</div> <div>text</div> </div>			
1	Moderate Intensity		
2	Easy Intensity		
3	Effort intensity		
4	Rest		
Total rows: 4 of 4		Query complete 00:00:00.038	Ln 1

✓ Successfully run. Total query runtime: 38 msec. 4 rows affected. ✕

17. Create view on patient table with check constraint condition.

Query:

```
--creating view
```

create view view_patient_table as

select * from "Patients"

where "Diabetes_Duration">10

with check option;

Data Output	Messages	Notifications
CREATE VIEW		
Query returned successfully in 39 msec.		
Total rows: 21 of 21	Query complete 00:00:00.039	✓ Query returned successfully in 39 msec.

--verifying the check option

insert into view_patient_table ("Patient_ID","Firstname","Lastname","Diabetes_Duration")

values ('s00','test','test','2');

Data Output	Messages	Notifications
ERROR: duplicate key value violates unique constraint "Patients_pkey"		
DETAIL: Key ("Patient_ID")=(s00) already exists.		
SQL state: 23505		
Total rows: 21 of 21	Query complete 00:00:00.090	

--calling the view

select * from view_patient_table;

Data Output

Messages

Notifications

	Patient_ID text	Firstname text	Lastname text	Visit_Date timestamp without time zone	Group_ID text	Gender_ID text	Race_ID text	BP_ID text	Opthal_ID text	WalkTest_ID text	Link_Reference_ID text	Age integer	Height real	BMI real	Diabetes_Duration integer
1	S0105	Jerrilyn	Kane	2005-09-29 12:46:00	GRP_02	G001	R001	BP006	OP006	WT006	LR006	54	1.78	33.96	
2	S0264	Far	Pardon	2005-09-30 09:24:00	GRP_02	G002	R001	BP016	OP016	WT016	LR016	66	1.72	27.41	
3	S0296	Sheena	Kevin	2005-12-24 09:15:00	GRP_02	G002	R001	BP018	OP018	WT018	LR018	71	1.56	22.68	
4	S0301	Daugherty	Conner	2005-11-27 01:18:00	GRP_02	G002	R002	BP019	OP019	WT019	LR019	65	1.63	34.7	
5	S0513	Ted	Scott	2005-06-09 07:33:00	GRP_02	G001	R001	BP029	OP029	WT029	LR029	61	1.58	40.02	
6	S0515	Ala	Halpert	2005-08-09 20:48:00	GRP_02	G002	R001	BP030	OP030	WT030	LR030	72	1.81	21.86	

Total rows: 22 of 22

Query complete 00:00:00.045

✓ Successfully run. Total query runtime: 45 msec. 22 rows affected.

18. Calculate the patient's current age.

Query:

```
Select concat ("Firstname" || ' ' || "Lastname")As "Fullname",((
Extract (Year from current_date)- Extract (Year from "Visit_Date"))+"Age")
as current_age from "Patients";
```

Data Output Messages Notifications		
	Fullname text	current_age numeric
1	Lanni Sue	77
2	Far Myers	78
3	Devlin Michael	92
4	Carmine Texas	84

Total rows: 77 of 77 Query complete 00:00:00.035

✓ Successfully run. Total query runtime: 35 msec. 77 rows affected. ✕

19. Write a query to display Mr. or Ms. as prefix to patients' names with respect to gender.

Query:

```
Select Case WHEN b."Gender" = 'Male'
THEN concat ('Mr.' || a."Firstname" || ' ' || a."Lastname")
when b."Gender" = 'Female'
THEN concat ('Mrs.' || a."Firstname" || ' ' || a."Lastname")
END AS PREFIX from
"Patients" a inner join "Gender" b
on a."Gender_ID" = b."Gender_ID";
```

Data Output Messages Notifications		
	prefix text	
1	Mrs.Lanni Sue	
2	Mr.Far Myers	
3	Mr.Devlin Michael	
4	Mrs.Carmine Texas	

Total rows: 77 of 77 Query complete 00:00:00.032

✓ Successfully run. Total query runtime: 32 msec. 77 rows affected. ✕

20. Write a query to get DM patient names whose distance is greater than 400 and speed is greater than 1.

Query:

```

select concat (a."Firstname" || ' ' || a."Lastname")As "Fullname"
from "Patients"a
inner join "Group"b
on a."Group_ID"= b."Group_ID"
inner join "Walking_Test"c
on a."WalkTest_ID" = c."WalkTest_ID"
WHERE b."Group"='DM' and c."Gait_DT_Distance" > 400 and c."Gait_DT_Speed" > 1;

```

Data Output

Messages

Notifications

	Fullname text
1	Jerrilyn Kane
2	Far Pardon
3	Gerhold Kendall
4	Bailey King

Total rows: 7 of 7

Query complete 00:00:00.031

✓ Successfully run. Total query runtime: 31 msec. 7 rows affected.

21. Create a trigger to raise notice and prevent the deletion of a record from a view.

Query:

Create view for patient table

```

CREATE VIEW PatientsView
AS
SELECT pnt."Patient_ID",pnt."Firstname", pnt."Lastname", pnt."Visit_Date",pnt."Age", pnt."Height",
pnt."BMI",
grp."Group",gnr."Gender",race."Race",bp."24Hr_Day_SBP",bp."24Hr_Day_DBP",bp."24Hr_Day_HR",
opt."Diabetic_Retinopathy", opt."Macular_Edema", lr."Lipid_ID", lr."Lab_ID"
FROM public."Patients" AS pnt
INNER JOIN public."Group" AS grp ON pnt."Group_ID" = grp."Group_ID"
INNER JOIN public."Gender" AS gnr ON pnt."Gender_ID" = gnr."Gender_ID"
INNER JOIN public."Race" AS race ON pnt."Race_ID" = race."Race_ID"
INNER JOIN public."Blood_Pressure" AS bp ON pnt."BP_ID" = bp."BP_ID"
INNER JOIN public."Ophthalmology" AS opt ON pnt."Opthal_ID" = opt."Opthal_ID"
INNER JOIN public."Link_Reference" AS lr ON pnt."Link_Reference_ID" = lr."Link_Reference_ID"

```

Create trigger to prevent delete record from view

```

CREATE TRIGGER TG_PATIENT_PREVENT_DELETE
INSTEAD OF DELETE

```



```

ON PatientsView
FOR EACH ROW
EXECUTE PROCEDURE patient_prevent_delete();

CREATE FUNCTION patient_prevent_delete()
RETURNS TRIGGER
AS $$
BEGIN
RAISE EXCEPTION 'You cannot delete records from View';
END;
$$
LANGUAGE plpgsql;

```

--Test Delete record operation--

```
delete from patientsview where "Patient_ID"='S0030';
```

```
108 delete from patientsview where "Patient_ID"='S0030';
```

Data Output Messages Notifications

```

ERROR: You cannot delete records from View
CONTEXT: PL/pgSQL function patient_prevent_delete() line 3 at RAISE
SQL state: P0001

```

22. Select the patient's full name with a name starting with 's' followed by any character, followed by 'r', followed by any character, followed by b.

Query:

```

Select Concat("Firstname",',','Lastname") As Fullname
from public."Patients"
where "Firstname" like 's_r_b';

```

Data Output

Messages

Notifications

fullname

text

Total rows: 0 of 0

Query complete 00:00:00.073

✓ Successfully run. Total query runtime: 73 msec. 0 rows affected.

23. write a query to get which race has the maximum number of Diabetic patients.

Query:

```
select R."Race" Race_with_Max_DMpatients,count(pt."Patient_ID") Number_Of_DiabeticPatients from
"Patients" pt
```

```
join "Race" R on R."Race_ID"=pt."Race_ID"
```

```
join "Group" Gr on pt."Group_ID" =Gr."Group_ID" where Gr."Group"='DM'
```

group by

R."Race" order by count(pt."Patient_ID") desc limit

	Data Output	Messages	Notifications
	+	📄	⌵
	race_with_max_dmpatients text	number_of_diabeticpatients bigint	
1	White	27	

Total rows: 1 of 1 Query complete 00:00:00.043

✓ Successfully run. Total query runtime: 43 msec. 1 rows affected.

Ln 86, Col 1

24. Identify the patient count by Gender and Race combination.

Query:

```
SELECT gnr."Gender",race."Race", count(pnt."Patient_ID")
FROM public."Patients" AS pnt
INNER JOIN public."Gender" AS gnr ON pnt."Gender_ID" = gnr."Gender_ID"
INNER JOIN public."Race" AS race ON pnt."Race_ID" = race."Race_ID"
group by gnr."Gender",race."Race" order by gnr."Gender",race."Race";
```

Data Output Messages Notifications			
	Gender text	Race text	count bigint
1	Female	African American	7
2	Female	Latino	2
3	Female	White	31
4	Male	African American	8
5	Male	Latino	1
6	Male	White	28

Total rows: 6 of 6 Query complete 00:00:00.044

✓ Successfully run. Total query runtime: 44 msec. 6 rows affected. ✕

25. Get the number of patients in the year 2005 in each of the Genesis and Cultivate labs.

Query:

```
SELECT lv."Lab_names", count(pnt."Patient_ID")
FROM public."Patients" pnt
INNER JOIN public."Link_Reference" lr ON pnt."Link_Reference_ID" = lr."Link_Reference_ID"
INNER JOIN public."Lab_Visit" as lv ON lr."Lab_visit_ID" = lv."Lab_visit_ID"
AND "Lab_names" IN ('Cultivate Lab','Genesis Lab')
AND date_part('year', "Lab_Visit_Date")::integer = '2005'
group by "Lab_names";
```

Data Output Messages Notifications		
	Lab_names text	count bigint
1	Cultivate Lab	8
2	Genesis Lab	6

Total rows: 2 of 2 Query complete 00:00:00.047

✓ Successfully run. Total query runtime: 47 msec. 2 rows affected. ✕

26. Write a query to get a list of patient IDs' and their Fasting Cholesterol in February 2006.

Query:

```
SELECT pnt."Patient_ID", lft."Fasting_Cholesterol"
FROM public."Patients" pnt
```

INNER JOIN public."Link_Reference" lr ON pnt."Link_Reference_ID" = lr."Link_Reference_ID"

INNER JOIN public."Lipid_Lab_Test" as llt ON lr."Lipid_ID" = llt."Lipid_ID"

INNER JOIN public."Lab_Visit" as lv ON lr."Lab_visit_ID" = lv."Lab_visit_ID"

AND to_char("Lab_Visit_Date", 'YYYY-MM') = '2006-02';

Data Output Messages Notifications		
Patient_ID	Fasting_Cholesterol	
text	integer	
1	S0033	151
2	S0551	168
3	S0610	121

Total rows: 3 of 3 Query complete 00:00:00.039

✓ Successfully run. Total query runtime: 39 msec. 3 rows affected. ✕

SELECT CONCAT ("Firstname", ' ', "Lastname") Patients_Name FROM public."Patients" where "Firstname"~*'^t';

Data Output Messages Notifications		
patients_name		
text		
1	Tann Bama	
2	Ted Scott	
3	Ted Stanley	
4	Ted Hue	
5	Tyler Nock	
6	Tony Polini	
7	Trent Kalil	

Total rows: 7 of 7 Query complete 00:00:00.057

✓ Successfully run. Total query runtime: 57 msec. 7 rows affected. ✕

Ln 94, Col 1

27. Write a query to get a list of patients whose first names is starting with the letter T

Query:

SELECT CONCAT ("Firstname", ' ', "Lastname") Patients_Name FROM public."Patients" where "Firstname"~*'^t';

Data Output		Messages	Notifications
<div> <div>patients_name</div> <div>text</div> </div>			
1	Tann Bama		
2	Ted Scott		
3	Ted Stanley		
4	Ted Hue		
5	Tyler Nock		
6	Tony Polini		
7	Trent Kalil		
Total rows: 7 of 7		Query complete 00:00:00.057	Ln 94, Col 1

✓ Successfully run. Total query runtime: 57 msec. 7 rows affected. ✕

28. Find a list of Male patients whose age is more than 60 whose, BMI is more than 18.5, and whose height is more than e 1.5 M.

Query:

```
SELECT CONCAT("Firstname", ' ', "Lastname") AS "MalePatients",a."Age",a."BMI",a."Height"
FROM public."Patients" AS a INNER JOIN public."Gender" AS b ON a."Gender_ID"=b."Gender_ID"
where b."Gender"='Male' and a."Age">60 and a."BMI">18.5 and a."Height">1.5;
```

Data Output

Messages

Notifications

MalePatients

text

Age

integer

BMI

real

Height

real

1

Far Myers

61

29.94

1.8

2

Devlin Michael

74

24.53

1.7

3

Bailey King

64

25.93

1.63

4

Walter Time

77

28.01

1.83

5

Gabriel Joseph

74

23.63

1.69

6

Sally Bing

63

26.54

1.7

7

Ted Scott

61

40.02

1.58

Total rows: 23 of 23

Query complete 00:00:00.051

✓ Successfully run. Total query runtime: 51 msec. 23 rows affected. ✕

Ln 90, Col 1

✓ Successfully run. Total query runtime: 51 msec. 23 rows affected. ✕

29.Display the patient names and ages whose age is more than 60 years.

Query:

```
SELECT CONCAT ("Firstname", ' ', "Lastname") AS "Fullname","Age"
FROM public."Patients" WHERE "Age">60;
```

Data Output Messages Notifications		
	Fullname text	Age integer
1	Far Myers	61
2	Devlin Michael	74
3	Carmine Texas	66
4	Tann Bama	72
5	Zonnya Ab	69
6	Far Pardon	66
7	Gerhold Kendall	66

✓ Successfully run. Total query runtime: 49 msec. 42 rows affected. ✕

30. Write a query to get the number of patients who visited the Lab between 9 am to 12 am.

Query:

```
SELECT count(pnt."Patient_ID")
FROM public."Patients" pnt
INNER JOIN public."Link_Reference" lr ON pnt."Link_Reference_ID" = lr."Link_Reference_ID"
INNER JOIN public."Lab_Visit" as lv ON lr."Lab_visit_ID" = lv."Lab_visit_ID"
AND date_part('hours', "Lab_Visit_Date")::integer BETWEEN 9 and 12;
```

Data Output Messages Notifications		
	count_of_patients bigint	
1	33	

✓ Successfully run. Total query runtime: 45 msec. 1 rows affected. ✕

31. Write a trigger that calls a function, for checking space and case for two columns or more before you add new data to a table.

Query:

```
drop table IF EXISTS Public."Patient_Dob";
```

Data Output
Messages
Notifications

DROP TABLE

Query returned successfully in 32 msec.

Total rows: 0 of 0
Query complete 00:00:00.032
✓ Query returned successfully in 32 msec. ✕

CREATE TABLE Public."Patient_Dob" (
Patient_Dob_id SERIAL PRIMARY KEY,
Patient_ID VARCHAR(100) NOT NULL,
DOB VARCHAR(100) NOT NULL,
Patient_Name VARCHAR(100) NULL
);

Data Output
Messages
Notifications

CREATE TABLE

Query returned successfully in 49 msec.

Total rows: 0 of 0
Query complete 00:00:00.049
✓ Query returned successfully in 49 msec. ✕

-- Creating function to remove the space
CREATE OR REPLACE FUNCTION Patient_Dob_fn()
RETURNS trigger AS \$\$
BEGIN
NEW.DOB:= TRIM(trailing from NEW.DOB);
NEW.Patient_Name:= TRIM(trailing from NEW.Patient_Name);
RETURN NEW;
END
\$\$ LANGUAGE plpgsql;

Data Output
Messages
Notifications

CREATE FUNCTION

Query returned successfully in 32 msec.

Total rows: 0 of 0
Query complete 00:00:00.032
✓ Query returned successfully in 32 msec. ✕

-- Creating Trigger for Patient_Dob to call the function (Patient_Dob_fn) to remove space
CREATE TRIGGER Patient_Dob_Trigger
BEFORE INSERT OR UPDATE on Public."Patient_Dob"
FOR EACH ROW
WHEN (NEW.DOB like '% ' or NEW.Patient_Name like '% ') EXECUTE PROCEDURE
public.Patient_Dob_fn();

Data Output
Messages
Notifications

CREATE TRIGGER
Query returned successfully in 49 msec.

Total rows: 0 of 0
Query complete 00:00:00.049
Query returned successfully in 49 msec.

INSERT INTO public."Patient_Dob"(
patient_id, dob, patient_name)
VALUES ("S0064", '1994 ', 'Piruthiha ');

Data Output
Messages
Notifications

INSERT 0 1
Query returned successfully in 48 msec.

Total rows: 0 of 0
Query complete 00:00:00.048
Query returned successfully in 48 msec.

SELECT * FROM "Patient_Dob"

Data Output
Messages
Notifications

	patient_dob_id [PK] integer	patient_id character varying (100)	dob character varying (100)	patient_name character varying (100)
1	1	"S0064"	1994	Piruthiha

Total rows: 1 of 1
Query complete 00:00:00.059
Successfully run. Total query runtime: 59 msec. 1 rows affected.

32. Display the first 6 characters of the lab names.

Query:

Select left ("Lab_names",6) as first_6_character from public."Lab_Visit";

Data Output		Messages	Notifications
		<div> <div>first_6_character</div> <div>text</div> </div>	
1	The Br		
2	Stream		
3	GreenD		
4	Cultiv		
5	Cultiv		
6	Cultiv		
7	Cultiv		
8	Stream		
9	Genesi		
10	Optima		
11	Optima		
12	GreenD		
13	The Br		
Total rows: 77 of 77		<div>Query complete 00:00:00.069</div> <div>Ln 9, Col 74</div>	

33. Write a query to create a table to get patients' demographic details whose birth year is 1939. Name the table as "Patient_Detail"

Query:

```
CREATE TABLE Public."Patient_Detail" (
Patient_Detail_id SERIAL PRIMARY KEY,
Patient_ID VARCHAR(100) NOT NULL,
Birthyear VARCHAR(100) NOT NULL,
Patient_Name VARCHAR(100) NULL
);
```

Data Output		Messages	Notifications
		<div>CREATE TABLE</div> <div>Query returned successfully in 52 msec.</div>	
Total rows: 77 of 77		<div>Query complete 00:00:00.052</div> <div>Query returned successfully in 52 msec.</div>	

```
INSERT INTO public."Patient_Detail"(
patient_id, birthyear, patient_name)
SELECT patient_id,date_part('YEAR',dt), "Patient_Name" FROM
(
SELECT "Patients"."Patient_ID" as "patient_id", now()::date - "Patients"."Age" *365 AS
"dt",concat("Firstname", "Lastname") as "Patient_Name"
FROM "Patients"
) AS FOO
where date_part('YEAR',dt) = 1939;
```

Data Output Messages Notifications

INSERT 0 0

Query returned successfully in 31 msec.

Total rows: 54 of 54 Query complete 00:00:00.031

✓ Query returned successfully in 31 msec. ✕

select * from public."Patient_Detail"

Data Output Messages Notifications

patient_detail_id	patient_id	birthyear	patient_name
[PK] integer	character varying (100)	character varying (100)	character varying (100)

Total rows: 0 of 0 Query complete 00:00:00.048

✓ Successfully run. Total query runtime: 48 msec. 0 rows affected. ✕

34. Write a query to get the number of patients above age 50 in each group

Query:

```
select gp."Group", count(*) Patients_Count from "Patients" pt
join "Group" gp on pt."Group_ID "=gp."Group_ID"
where pt."Age">50
group by gp."Group"
```

26

Data Output Messages Notifications

Group	patients_count
CONTROL	33
DM	42

Total rows: 2 of 2 Query complete 00:00:00.044

✓ Successfully run. Total query runtime: 44 msec. 2 rows affected. ✕

35. Write a query to find the number of patients visited each month. (Display with month Name)

Query:

```
SELECT TO_CHAR(to_date(CAST("Visit_Date" AS
TEXT),'YYYY-MM-DD'),'Month') AS Month_name,
COUNT(*) As No_Patients
FROM "Patients"
Group by 1;
```

Data Output			Messages	Notifications
	month_name text	no_patients bigint		
1	January	4		
2	February	8		
3	November	4		
4	September	6		
5	August	10		
6	October	14		
7	December	5		
8	June	7		
9	March	3		
10	May	4		
11	April	4		
12	July	8		

Total rows: 12 of 12 Query complete 00:00:00.045

✓ Successfully run. Total query runtime: 45 msec. 12 rows affected. ✕

36. Write a query to get a number of visual/motor dementia patients who have any 2 abnormal conditions. (Display with condition name). (dementia/cognitive impairment: any patient who has any two abnormal test results).

Query:

```
select pt."Patient_ID", concat(pt."Firstname",pt."Lastname") as "Patient_Name" ,
case when (vl."RCFT_IR"<=71 and vl."Clock"<=2) then 'person has chance of dementia(RCFT_IR<=71 and
clock<=2)'
when (vl."RCFT_IR"<=71 and vl."TM">=42) then 'person has Dementia/Cognitive Impairment
(RCFT_IR<=71 and TM>=42)' end as "Condition"
from
public. "Patients" pt
join "Link_Reference" lr on pt."Link_Reference_ID"=lr."Link_Reference_ID"
join "Visual/Motor_Cog" vl on vl."VM_ID"=lr."VM_ID"
where (vl."RCFT_IR"<=71 and vl."Clock"<=2) or (vl."RCFT_IR"<=71 and vl."TM">=42)
order by vl."Clock" asc
```

Data Output Messages Notifications			
	Patient_ID [PK] text	Patient_Name text	Condition text
1	S0582	RosieJakubowski	person has chance of dementia(RCFT_IR<=71 and clock<=2)
2	S0570	PamelaBonco	person has Dementia/Cognitive Impairment (RCFT_IR<=71 and TM>=4...
3	S0574	TonyPolini	person has Dementia/Cognitive Impairment (RCFT_IR<=71 and TM>=4...
4	S0544	RyanAngear	person has Dementia/Cognitive Impairment (RCFT_IR<=71 and TM>=4...
5	S0068	CarmineTexas	person has Dementia/Cognitive Impairment (RCFT_IR<=71 and TM>=4...
6	S0561	BrendaPolini	person has Dementia/Cognitive Impairment (RCFT_IR<=71 and TM>=4...
7	S0597	LolitaDarci	person has Dementia/Cognitive Impairment (RCFT_IR<=71 and TM>=4...
8	S0560	BridgetPhilipsson	person has Dementia/Cognitive Impairment (RCFT_IR<=71 and TM>=4...
9	S0557	MikeAnthill	person has Dementia/Cognitive Impairment (RCFT_IR<=71 and TM>=4...
10	S0601	KellenWandle	person has Dementia/Cognitive Impairment (RCFT_IR<=71 and TM>=4...
11	S0584	ErikGardie	person has Dementia/Cognitive Impairment (RCFT_IR<=71 and TM>=4...
12	S0411	GabrielJoseph	person has Dementia/Cognitive Impairment (RCFT_IR<=71 and TM>=4...
Total rows: 54 of 54		Query complete 00:00:00.039	

✓ Successfully run. Total query runtime: 39 msec. 54 rows affected. ✕

37. Write a query to get a list of patient IDs whose fasting glucose is 80, 85, and 89.

Query:

SELECT "Patient_ID","Fasting_Glucose" from public."Lab_Test"

where "Fasting_Glucose" in (80,85,89)

Data Output Messages Notifications			
	Patient_ID text	Fasting_Glucose integer	
1	S0030	89	
2	S0064	85	
3	S0228	89	
4	S0454	85	
5	S0527	80	
6	S0541	80	
7	S0579	89	
8	S0595	80	
9	S0600	89	
Total rows: 9 of 9		Query complete 00:00:00.114	

✓ Successfully run. Total query runtime: 114 msec. 9 rows affected. ✕

38. calculate the difference between Day and night HR. (Display 2 decimal only)

Query:

select round(cast(("24Hr_Day_HR"-"24Hr_Night_HR") as decimal),2) DayNight_HR_difference

from public."Blood_Pressure"

Data Output		Messages	Notifications
daynight_hr_difference	numeric		
1	8.74		
2	7.81		
3	8.90		
4	10.21		
5	13.24		
6	9.13		
7	18.17		
8	3.77		
9	3.00		
10	6.17		
11	11.19		
12	7.52		
13	9.91		
Total rows: 77 of 77		Query complete 00:00:00.042	✓ Successfully run. Total query runtime: 42 msec. 77 rows affected.

39. Find out the tables where column Patient_ID is present. (Display column position number with respective table also)

Query:

```
SELECT ORDINAL_POSITION AS column_position, tables.table_name
FROM information_schema.tables AS tables
JOIN information_schema.columns AS columns
ON columns.table_name = tables.table_name
AND columns.table_schema = tables.table_schema
WHERE columns.column_name = 'Patient_ID'
ORDER BY tables.table_name;
```

Data Output		Messages	Notifications
column_position	table_name		
integer	name		
1	2 Blood_Pressure		
2	2 Lab_Test		
3	2 Lipid_Lab_Test		
4	1 Patients		
5	2 Verbal_Cognitive		
6	2 Visual/Motor_Cog		
7	2 Walking_Test		
Total rows: 8 of 8		Query complete 00:00:00.142	✓ Successfully run. Total query runtime: 142 msec. 8 rows affected. ✕

40. Display the first name and Last name of patients whose Race is Latino.

Query:

```
Select P."Firstname",P."Lastname",R."Race"
from public."Patients" P,public."Race" R
Where P."Race_ID"=R."Race_ID"
and R."Race"='Latino'
```

Data Output Messages Notifications			
	Firstname text	Lastname text	Race text
1	Daugherty	Conner	Latino
2	Joeph	Long	Latino
3	Ryan	Angear	Latino

Total rows: 3 of 3 Query complete 00:00:00.051

✓ Successfully run. Total query runtime: 51 msec. 3 rows affected. X

41. write a query to get the number of patients whose urine creatinine is in a normal range (Gender wise).

Query:

```
select G."Gender",count(P."Patient_ID") from "Patients" P
inner join "Link_Reference" LF on LF."Link_Reference_ID"=P."Link_Reference_ID"
inner join "Urine_Test" UT on UT."Urine_ID"=LF."Urine_ID"
inner join "Gender" G on G."Gender_ID"=P."Gender_ID"
where UT."Creatinine" between 65.4 and 119.3 and G."Gender"='Male'
group by G."Gender"
union
select G."Gender",count(P."Patient_ID") from "Patients" P
inner join "Link_Reference" LF on LF."Link_Reference_ID"=P."Link_Reference_ID"
inner join "Urine_Test" UT on UT."Urine_ID"=LF."Urine_ID"
inner join "Gender" G on G."Gender_ID"=P."Gender_ID"
where UT."Creatinine" between 52.2 and 91.9 and G."Gender"='Female'
```

group by G."Gender"

Data Output			Messages	Notifications
<div> <div> <div>≡</div> <div>+</div> </div> <div> <div>📄</div> <div>▼</div> </div> <div> <div>📄</div> <div>🗑️</div> </div> <div> <div>📄</div> <div>📄</div> </div> <div> <div>📄</div> <div>📄</div> </div> <div> <div>📄</div> <div>📄</div> </div> <div> <div>📄</div> <div>📄</div> </div> </div>				
	Gender	count		
	text	bigint		
1	Female	19		
2	Male	17		

Total rows: 2 of 2
Query complete 00:00:00.033

✓ Successfully run. Total query runtime: 33 msec. 2 rows affected.
✕

42. Write a query to update id LB002 with the lab name Cultivate Lab.

Query:

UPDATING:

```

update "Lab_Visit" LV set "Lab_names"='Cultivate Lab'
from "Link_Reference" LR
inner join "Lab_Test" LT on LT."Lab_ID"=LR."Lab_ID"
where LR."Lab_visit_ID" = LV."Lab_visit_ID" and LT."Lab_ID"='LB002'

```

Data Output			Messages	Notifications
<div> <div> <div>≡</div> <div>+</div> </div> <div> <div>📄</div> <div>▼</div> </div> <div> <div>📄</div> <div>🗑️</div> </div> <div> <div>📄</div> <div>📄</div> </div> <div> <div>📄</div> <div>📄</div> </div> <div> <div>📄</div> <div>📄</div> </div> <div> <div>📄</div> <div>📄</div> </div> </div>				
UPDATE 1				
Query returned successfully in 34 msec.				

Total rows: 2 of 2
Query complete 00:00:00.034

✓ Query returned successfully in 34 msec.
✕

VIEWING THE RESULT:

```

select LV.*
from "Lab_Visit" LV
inner join "Link_Reference" LR on LR."Lab_visit_ID" = LV."Lab_visit_ID"
inner join "Lab_Test" LT on LT."Lab_ID"=LR."Lab_ID"
where LT."Lab_ID"='LB002';

```

Data Output	Messages	Notifications
<div> <div> <div></div> <div></div> <div></div> <div></div> <div></div> <div></div> <div></div> </div> <div> <div></div> <div></div> <div></div> <div></div> <div></div> <div></div> <div></div> </div> </div>	<div> <div>Lab_visit_ID</div> <div>[PK] text</div> </div> <div> <div>Lab_Visit_Date</div> <div>timestamp without time zone</div> </div> <div> <div>Lab_names</div> <div>text</div> </div>	<div> <div>1</div> <div>LV002</div> <div>2006-02-14 11:58:55</div> <div>Cultivate Lab</div> </div>
<div> <div>Total rows: 1 of 1</div> <div>Query complete 00:00:00.041</div> <div> <div>✓ Successfully run. Total query runtime: 41 msec. 1 rows affected.</div> <div>✗</div> </div> </div>		

43.Create an index on any table and use explain analyze to show differences if any.

Query:

```
CREATE INDEX idx_Patient_ID ON "Patients"("Patient_ID");
```

```
EXPLAIN (ANALYZE) SELECT * from public."Patients"
```

Data Output	Messages	Notifications
<div> <div> <div></div> <div></div> <div></div> <div></div> <div></div> <div></div> <div></div> </div> <div> <div></div> <div></div> <div></div> <div></div> <div></div> <div></div> <div></div> </div> </div>	<div> <div>QUERY PLAN</div> <div>text</div> </div> <div> <div>1</div> <div>Seq Scan on "Patients" (cost=0.00..2.77 rows=77 width=83) (actual time=0.024..0.032 rows=77 loops...</div> </div> <div> <div>2</div> <div>Planning Time: 2.024 ms</div> </div> <div> <div>3</div> <div>Execution Time: 0.051 ms</div> </div>	<div> <div>Total rows: 3 of 3</div> <div>Query complete 00:00:00.062</div> <div> <div>✓ Successfully run. Total query runtime: 62 msec. 3 rows affected.</div> <div>✗</div> </div> </div>

44.Write a query to split the lab visit date into two different columns lab_visit_date and lab_visit_time.

Query:

```
Select P."Firstname",P."Lastname",cast(LV."Lab_Visit_Date" as Date) as "lab_visit_date",
cast(LV."Lab_Visit_Date" as time) "lab_visit_time"
```

```
from "Lab_Visit" LV
```

```
inner join "Link_Reference" LR on LR."Lab_visit_ID" = LV."Lab_visit_ID"
```

```
inner join "Lab_Test" LT on LT."Lab_ID"=LR."Lab_ID"
```

```
inner join "Patients" P on P."Link_Reference_ID" =LR."Link_Reference_ID"
```


Data Output Messages Notifications				
	Firstname text	Lastname text	lab_visit_date date	lab_visit_time time without time zone
1	Lanni	Sue	2006-06-12	12:18:18
2	Far	Myers	2006-02-14	11:58:55
3	Devlin	Michael	2005-02-21	10:57:43
Total rows: 77 of 77 Query complete 00:00:00.043				

✓ Successfully run. Total query runtime: 43 msec. 77 rows affected. ✕

45. Please go through the below screenshot and create the exact output

Query:

```
select cast(translate("Patient_ID",'S','') as int) as "pat_id", bool_and((cast((translate("Patient_ID",'S','')) as int) % 2) = 0) as "even" ,
```

```
bool_and((cast((translate("Patient_ID",'S','')) as int) % 2) != 0) as "odd"
```

```
from public."Patients"
```

```
group by "Patient_ID"
```

Data Output Messages Notifications				
	pat_id integer	even boolean	odd boolean	
1	565	false	true	
2	569	false	true	
3	454	true	false	
Total rows: 77 of 77 Query complete 00:00:00.034				

✓ Successfully run. Total query runtime: 34 msec. 77 rows affected. ✕

46. Calculate the Number of Diabetic Male and Female patients who are Anemic.

Query:

```
select G."Gender",count(GP."Group") "Diabetic_Count" from "Patients" P
```

```
inner join "Gender" G on G."Gender_ID"=P."Gender_ID"
```

```
inner join "Lab_Test" LT on LT."Patient_ID"=P."Patient_ID"
```

```
inner join "Group" GP on GP."Group_ID"=P."Group_ID" and G."Gender"='Male'
```

```

where GP."Group"='DM' and LT."Hgb" < 13.2
group by G."Gender"
union
select G."Gender",count(GP."Group") "Diabetic_Count" from "Patients" P
inner join "Gender" G on G."Gender_ID"=P."Gender_ID"
inner join "Lab_Test" LT on LT."Patient_ID"=P."Patient_ID"
inner join "Group" GP on GP."Group_ID"=P."Group_ID" and G."Gender"='Female'
where GP."Group"='DM' and LT."Hgb" < 11.6
group by G."Gender"

```

Data Output Messages Notifications		
Gender	Diabetic_Count	
text	bigint	
1	Female	3
2	Male	14

Total rows: 2 of 2 Query complete 00:00:00.036

✓ Successfully run. Total query runtime: 36 msec. 2 rows affected. ✕

47. Write a query to display the Patient_ID, last name, and the position of the substring 'an' in the last name column for those patients who have a substring 'an'.

Query:

```

select "Lastname" , position('an' in "Lastname") "Position of 'an'" from public."Patients" where
substring("Lastname",position('an' in "Lastname"),2)='an'

```

Data Output Messages Notifications		
Lastname	Position of 'an'	
text	integer	
1	Kane	2
2	Man	2
3	Stanley	3

Total rows: 6 of 6 Query complete 00:00:00.036

✓ Successfully run. Total query runtime: 36 msec. 6 rows affected. ✕

48. List of patients from rows 30-40 without using the where condition.

Query:

```
select * from (select row_number() OVER (ORDER BY "Patient_ID") AS "RowNum",* from public."Patients" ) fnl where fnl."RowNum" between 30 and 40
```

Data Output Messages Notifications											
	RowNum bigint	Patient_ID [PK] text	Firstname text	Lastname text	Visit_Date timestamp without time zone	Group_ID text	Gender_ID text	Race_ID text	BP_ID text	Optha_ID text	WalkTest_ID text
1	30	S0515	Ala	Halpert	2005-08-09 20:48:00	GRP_02	G002	R001	BP030	OP030	WT030
2	31	S0516	Harry	Ding	2005-02-23 02:46:00	GRP_01	G002	R001	BP031	OP031	WT031
3	32	S0522	Barry	Creed	2005-10-20 02:23:00	GRP_02	G001	R001	BP032	OP032	WT032
4	33	S0523	Ted	Stanley	2005-06-05 06:05:00	GRP_01	G001	R001	BP033	OP033	WT033
5	34	S0527	Ted	Hue	2006-07-04 06:46:00	GRP_02	G001	R003	BP034	OP034	WT034
6	35	S0531	Fred	Tye	2006-08-07 13:52:00	GRP_02	G001	R001	BP035	OP035	WT035
7	36	S0532	Kim	Cone	2005-12-22 21:21:00	GRP_01	G001	R001	BP036	OP036	WT036
8	37	S0534	Sarah	West	2005-09-01 02:06:00	GRP_02	G001	R003	BP037	OP037	WT037
9	38	S0536	Abigail	Renter	2005-10-01 21:35:00	GRP_02	G001	R003	BP038	OP038	WT038
10	39	S0539	Dave	D'Ambrogi	2005-11-10 20:49:00	GRP_02	G001	R001	BP039	OP039	WT039
11	40	S0540	Christian	Pow	2005-07-28 17:03:00	GRP_02	G001	R001	BP040	OP040	WT040

Total rows: 11 of 11 Query complete 00:00:00.048

✓ Successfully run. Total query runtime: 48 msec. 11 rows affected. ✕

49. Find all patients whose last name contains the text either Mc or Bu using Regular expression matches.

Query:

```
SELECT "Lastname" FROM public."Patients" WHERE "Lastname" ~ '^(Mc|Bu).*'
```

Data Output

Messages

Notifications

Lastname

text

Total rows: 0 of 0

Query complete 00:00:00.038

✓ Successfully run. Total query runtime: 38 msec. 0 rows affected.

✕

50. Create materialized view with no data, to display no of male and female patients.

Query:

```
CREATE MATERIALIZED VIEW Number_Of_Gender
```

```
AS
```

```
select G."Gender",count(G."Gender") "No_Of_Gender" from "Patients" P
```

```
inner join "Gender" G on G."Gender_ID"=P."Gender_ID"
```

```
group by G."Gender"
```

```
WITH NO DATA;
```

526

Data Output Messages Notifications

CREATE MATERIALIZED VIEW

Query returned successfully in 37 msec.

Total rows: 0 of 0 Query complete 00:00:00.037

✓ Query returned successfully in 37 msec.

REFRESH MATERIALIZED VIEW Number_Of_Gender;

526

Data Output Messages Notifications

CREATE MATERIALIZED VIEW

Query returned successfully in 37 msec.

Total rows: 0 of 0 Query complete 00:00:00.037

✓ Query returned successfully in 37 msec.

```
select * from Number_Of_Gender
```

Data Output Messages Notifications

	Gender text	No_Of_Gender bigint
1	Female	40
2	Male	37

Total rows: 2 of 2 Query complete 00:00:00.035

✓ Successfully run. Total query runtime: 35 msec. 2 rows affected. ✕

51. Get a list of unique lab names whose names is starting with G and end with b.

Query:

```
select distinct "Lab_names" from public."Lab_Visit" where "Lab_names" like 'G%b';
```

Data Output		Messages	Notifications
<div> <div>Lab_names</div> <div>text</div> <div>1 Genesis Lab</div> <div>2 GreenDots lab</div> </div>			
Total rows: 2 of 2		Query complete 00:00:00.099	
		<div> <div>✓ Successfully run. Total query runtime: 99 msec. 2 rows affected.</div> <div>X</div> </div>	
		Ln 60, Col 82	

52. Write the query to create an Index on table Verbal_Cognitive by selecting a column and also write the query drop the same index.

Query:

```
Create index Verb_index on public."Verbal_Cognitive" ("HVLt");
```

Data Output		Messages	Notifications
<div> <div>CREATE INDEX</div> <div>Query returned successfully in 68 msec.</div> </div>			
Total rows: 10 of 10		Query complete 00:00:00.068	
		<div> <div>✓ Query returned successfully in 68 msec.</div> <div>X</div> </div>	
		Ln 89, Col 63	

```
Drop index Verb_index;
```

Data Output	Messages	Notifications
DROP INDEX		
Query returned successfully in 99 msec.		

Total rows: 10 of 10	Query complete 00:00:00.099	Ln 91, Col 23
----------------------	-----------------------------	---------------

53. Get the number of patients born in a leap year.

Query:

```
CREATE OR REPLACE FUNCTION is_leap_year(year int)
RETURNS BOOLEAN AS $$
```

```
SELECT ($1 % 4 = 0) AND (($1 % 100 <> 0) or ($1 % 400 = 0))
```

```
$$ LANGUAGE sql IMMUTABLE STRICT;
```

```
Select count(*) as leapyear_born from public."Patients"
```

```
where is_leap_year(cast(EXTRACT (YEAR FROM CURRENT_DATE) as int) - "Age" )='true'
```

Data Output	Messages	Notifications
<div> <div>leapyear_born</div> <div>bigint</div> </div>		
1	11	
<div>Total rows: 1 of 1</div> <div>Query complete 00:00:00.037</div> <div>Successfully run. Total query runtime: 37 msec. 1 rows affected.</div>		

54. Write a query to get a list of patient IDs from the DM group and above age 60 in sequence.

Query:

```
Select a."Patient_ID" as DM_Patient_ID_above60, CONCAT (a."Firstname" || ' ' || a."Lastname")As
"Fullname", a."Age" from "Patients" a
```

Inner join "Group"b on a."Group_ID"=b."Group_ID"

where "Group"='DM'

And "Age">'60'

order by "Age" Asc;

Data Output Messages Notifications			
	dm_patient_id_above60 text	Fullname text	Age integer
1	S0513	Ted Scott	61
2	S0534	Sarah West	61
3	S0570	Pamela Bonco	62
4	S0580	Harry Florie	62
5	S0430	Lincoln Brenda	62
6	S0434	Sally Bing	63
Total rows: 26 of 26 Query complete 00:00:00.038			

✓ Successfully run. Total query runtime: 38 msec. 26 rows affected. ✕

55.Find the patient who has the most damage in the eyes with the use of a max function.

Query:

Select concat(a."Firstname" || ' ' || a."Lastname") from "Patients"a

join "Ophthalmology"b On a."Opthal_ID"= b. "Opthal_ID"

where b."Diabetic_Retinopathy" =(select max(b."Diabetic_Retinopathy") from "Ophthalmology"b);

Data Output Messages Notifications			
	concat text		
1	Jordan Ross		
2	Christian Pow		
3	Kent Priestnall		
4	Kellen Wandle		
Total rows: 4 of 4 Query complete 00:00:00.033			

✓ Successfully run. Total query runtime: 33 msec. 4 rows affected. ✕

56.Create a procedure for checking if Race exists using an if else statement.

Query:

```
CREATE or replace PROCEDURE Pkg_Race_IS_Exists(INOUT RaceExist varchar)
LANGUAGE plpgsql
AS $$
BEGIN
select 1 into RaceExist from "Race" where "Race" =RaceExist;

if(RaceExist='1') THEN
RaceExist='True';
ELSE
RaceExist='False';
end IF;
END;
$$;
```

Data Output Messages Notifications

CREATE PROCEDURE

Query returned successfully in 36 msec.

Total rows: 77 of 77 Query complete 00:00:00.036

✓ Query returned successfully in 36 msec. ✕

CALL public.pkg_race_is_exists('White')

Data Output Messages Notifications

raceexist
character varying

1	True
---	------

Total rows: 1 of 1 Query complete 00:00:00.051

✓ Successfully run. Total query runtime: 51 msec. 1 rows affected. ✕

57. Write a query to get a list of female patients who are at risk of heart diseases with the help of Fasting HDL.

Query:


```
select concat(pt."Firstname" || ' ' || pt."Lastname") Female_Heart_Disease_Risk from "Patients" pt
join "Gender" gn on pt."Gender_ID"=gn."Gender_ID"
join "Link_Reference" lr on pt."Link_Reference_ID"=lr."Link_Reference_ID"
join "Lipid_Lab_Test" llt on lr."Lipid_ID"=llt."Lipid_ID"
where gn."Gender"='Female' and llt."Fasting_HDL"<50
```

The screenshot shows a database query result in a web interface. The 'Data Output' tab is active, displaying a table with 20 rows. The first column is labeled 'female_heart_disease_risk' and the second column is labeled 'text'. The data includes names and locations of patients. A status bar at the bottom indicates 'Total rows: 20 of 20' and 'Query complete 00:00:00.043'. A green message box at the bottom right states 'Successfully run. Total query runtime: 43 msec. 20 rows affected.'

female_heart_disease_risk	text
1	Carmine Texas
2	Devlin Black
3	Dean Kimberly
4	Skipple Marriot
5	Far Pardon
6	Gerhold Kendall
7	Sheena Kevin
8	Daugherty Conner
9	Lincoln Brenda
...	...
20	...

58. Create a role via query.

Query:

```
CREATE ROLE Johnsmith with login PASSWORD '<password>';
```

The screenshot shows a database query result in a web interface. The 'Messages' tab is active, displaying the message 'CREATE ROLE'. Below it, a status bar indicates 'Query returned successfully in 50 msec.'. A green message box at the bottom right states 'Query returned successfully in 50 msec. X'. The bottom status bar shows 'Total rows: 3 of 3' and 'Query complete 00:00:00.050'.

Message
CREATE ROLE

59. Query to list all the users in the DB.

Query:

```
select * from pg_catalog.pg_user;
```

Data Output Messages Notifications										
	username name	usesysid oid	usecreatedb boolean	usesuper boolean	userepl boolean	usebypassrls boolean	passwd text	valuntil timestamp with time zone	useconfig text[]	
1	postgres	10	true	true	true	true	*****	[null]	[null]	
2	johnsmith	24794	false	false	false	false	*****	[null]	[null]	

Total rows: 2 of 2 Query complete 00:00:00.094

✓ Successfully run. Total query runtime: 94 msec. 2 rows affected. ✕

Ln 47, Col 34

60. Write a query using the trigger after insert on the lab test table. If the patient has abnormal HbA1C and fasting glucose values.

Query:

```
CREATE TRIGGER TG_Labtest
AFTER INSERT ON public."Lab_Test"
FOR EACH ROW
EXECUTE PROCEDURE labtest_abnormal_hba1c_glucose();
drop function labtest_abnormal_hba1c_glucose();
CREATE FUNCTION labtest_abnormal_hba1c_glucose()
RETURNS TRIGGER
AS $BODY$
BEGIN
IF "Hb_A1C" > 5.7 and (NEW."Fasting_Glucose" < 70 OR NEW."Fasting_Glucose" > 100) THEN
RAISE INFO 'Patinet has abnormal HbA1C and fasting glucose values';
END IF;
return new;
END;
$BODY$
LANGUAGE plpgsql;
```

Data Output	Messages	Notifications
CREATE TRIGGER		
Query returned successfully in 49 msec.		

61.write a query to get the number of patients for each age bin without using the CASE statement.(Bin size - 5)

Query:

```
--select width_bucket("Age",5,80,16),"Age" from "Patients"
```

```
select count(*) PatientsCount, concat (width_bucket("Age",5,80,16)||' '||'th' || ' '||'AgeBin')
agebin_Number from "Patients"
```

```
group by agebin_Number order by agebin_Number asc
```

--Explanation

--To create a age bin size of 5,

--'5' is given as the low value for the "width_bucket" function and '80' is the high value.

--So, to have equal bin size of 5(80 divided by 5=16) '16' is given as the number of bins to be created.

--in the result count of patients for each bin postion will be displayed. Eg, 10th bin means, (10*5=50) age bin 50

Data Output	Messages	Notifications																								
<div> <div> <div></div> <div></div> <div></div> <div></div> <div></div> <div></div> <div></div> <div></div> </div> <div> <div></div> <div></div> <div></div> <div></div> <div></div> <div></div> <div></div> <div></div> </div> </div>																										
<table> <thead> <tr> <th></th><th>agebin_number text</th><th>patientscount bigint</th></tr> </thead> <tbody> <tr><td>1</td><td>10th AgeBin</td><td>3</td></tr> <tr><td>2</td><td>11th AgeBin</td><td>13</td></tr> <tr><td>3</td><td>12th AgeBin</td><td>22</td></tr> <tr><td>4</td><td>13th AgeBin</td><td>9</td></tr> <tr><td>5</td><td>14th AgeBin</td><td>12</td></tr> <tr><td>6</td><td>15th AgeBin</td><td>13</td></tr> <tr><td>7</td><td>16th AgeBin</td><td>5</td></tr> </tbody> </table>				agebin_number text	patientscount bigint	1	10th AgeBin	3	2	11th AgeBin	13	3	12th AgeBin	22	4	13th AgeBin	9	5	14th AgeBin	12	6	15th AgeBin	13	7	16th AgeBin	5
	agebin_number text	patientscount bigint																								
1	10th AgeBin	3																								
2	11th AgeBin	13																								
3	12th AgeBin	22																								
4	13th AgeBin	9																								
5	14th AgeBin	12																								
6	15th AgeBin	13																								
7	16th AgeBin	5																								
<div>Total rows: 7 of 7</div> <div>Query complete 00:00:00.0039</div> <div>✓ Successfully run. Total query runtime: 39 msec. 7 rows affected.</div>																										

62. Write a query to get the number of patients who have normal platelets for each group.

Query:

```
select GP."Group_ID", count(P."Patient_ID") "patients_Count"
from "Patients" P
inner join "Gender" G on G."Gender_ID"=P."Gender_ID"
inner join "Lab_Test" LT on LT."Patient_ID"=P."Patient_ID"
inner join "Link_Reference" LR on LT."Lab_ID"=LR."Lab_ID"
inner join "Group" GP on GP."Group_ID"=P."Group_ID"
where LT."Platelets" between 150 and 450
group by GP."Group_ID"
```

Data Output Messages Notifications			
	Group_ID [PK] text	patients_Count bigint	
1	GRP_01	31	
2	GRP_02	40	

Total rows: 2 of 2 Query complete 00:00:00.037

✓ Successfully run. Total query runtime: 37 msec. 2 rows affected. ✕

63. Create a trigger on a view of the Blood Pressure table.

Query:

```
--Creating view for blood_pressure Table
CREATE OR REPLACE VIEW public.blood_pressure_view
AS SELECT * FROM "Blood_Pressure";
```

Data Output Messages Notifications			
CREATE VIEW			
Query returned successfully in 56 msec.			

Total rows: 4 of 4 Query complete 00:00:00.056

✓ Query returned successfully in 56 msec. ✕

```
-- Creating function
create or replace function blood_pressure_Fn() returns trigger as $blood_pressure_Fn$
begin
return null;
end;
$blood_pressure_Fn$ language plpgsql;
```

Data Output	Messages	Notifications
CREATE FUNCTION		
Query returned successfully in 36 msec.		

Total rows: 4 of 4	Query complete 00:00:00.036	✓ Query returned successfully in 36 msec. ✕
--------------------	-----------------------------	---

```
-- Creating Trigger for blood_pressure the blood_pressure_view
create trigger blood_pressure_trigger
instead of delete on blood_pressure_view
for each row execute procedure blood_pressure_Fn();
```

```
select * from blood_pressure_view
```

Data Output

Messages

Notifications

	BP_ID text	Patient_ID text	24Hr_Day_SBP real	24Hr_Day_DBP real	24Hr_Day_HR real	24Hr_Night_SBP real	24Hr_Night_DBP real	24Hr_Night_HR real	
1	BP001	S0030	133	74.9	75.12	118.31	64.42	66.38	
2	BP002	S0033	131.51	71.49	63.39	137.92	69.13	55.58	
3	BP003	S0064	130.79	64.03	58.26	116.56	53.8	49.36	
4	BP004	S0068	116.72	62.03	67.62	109.24	52	57.41	
5	BP005	S0078	134.53	61.26	75.24	131.41	56.82	62	
6	BP006	S0105	135.57	61.24	65.51	139.08	56.77	56.38	
Total rows: 77 of 77 Query complete 00:00:00.044									

✓ Successfully run. Total query runtime: 44 msec. 77 rows affected. ✕

64. Write a query to find the number of Patients whose Gait RPE start is greater than the end and vice versa. (Display exact output as shown below)

Query:

```
(select 'RPE_End>Start' as type, count(*) from "Walking_Test"b
```

```
inner join "Patients"a
on a."Patient_ID" = b."Patient_ID"
where "Gait_RPE_End " > "Gait_RPE_Start ")
```

union all

```
(SELECT 'RPE_Start>End' as type,count(*) FROM "Walking_Test"b
inner join "Patients"a
on b."Patient_ID" = a."Patient_ID"
where "Gait_RPE_Start " > "Gait_RPE_End ");
```

Data Output Messages Notifications			
	type	count	
	text	bigint	
1	RPE_End>Start	52	
2	RPE_Start>End	2	

Total rows: 2 of 2 Query complete 00:00:00.031

✓ Successfully run. Total query runtime: 31 msec. 2 rows affected. ✕

65.Create a view without using any schema or table and check the created view using a select statement.

Query:

```
create view "Dummy1" as select 10;
select * from "Dummy1";
```

Data Output	Messages	Notifications
<div> <div>?</div> <div>column?</div> <div>integer</div> </div> <div> <div>1</div> <div>10</div> </div>		
<div> <div>✓ Successfully run. Total query runtime: 52 msec. 1 rows affected. ✕</div> </div>		
<div> <div>Total rows: 1 of 1</div> <div>Query complete 00:00:00.052</div> <div>Ln 78, Col 24</div> </div>		

66. Display patients names who have the same last name

Query:

```
SELECT CONCAT ("Firstname" || ' ' || "Lastname")
FROM "Patients"
WHERE "Lastname" in (Select "Lastname"from "Patients"
GROUP BY "Lastname" having count ("Lastname")>1);
```

Data Output	Messages	Notifications
<div> <div>concat</div> <div>text</div> </div> <div> <div>1</div> <div>Sally Bing</div> </div> <div> <div>2</div> <div>Mike Bing</div> </div> <div> <div>3</div> <div>Christian Pow</div> </div> <div> <div>4</div> <div>Tyler Nock</div> </div>		
<div> <div>✓ Successfully run. Total query runtime: 32 msec. 10 rows affected. ✕</div> </div>		
<div> <div>Total rows: 10 of 10</div> <div>Query complete 00:00:00.032</div> <div>0</div> </div>		

67. Write a query to get the Sum of Diabetes Duration for Group id 'GRP_02'.

Query:

```
SELECT Sum ("Diabetes_Duration")
From "Patients" WHERE "Group_ID"= 'GRP_02';
```

Data Output	Messages	Notifications
<div> <div>sum</div> <div>bigint</div> <div>1</div> <div>532</div> </div>		
Total rows: 1 of 1 Query complete 00:00:00.034 ✓ Successfully run. Total query runtime: 34 msec. 1 rows affected. ✕		

68. Write a query to get a patient name who has a chance to have kidney disease with Albumin

Query:

```
SELECT CONCAT (a."Firstname" || ' ' || a."Lastname") As
"Patients_chance_to_have_KidneyDisease_with_Albumin"
from
"Patients" a inner join "Link_Reference" b
on a."Link_Reference_ID" = b."Link_Reference_ID"
inner join "Urine_Test" c
on b."Urine_ID" = C."Urine_ID"
WHERE "Albumin" >='30';
```

Data Output	Messages	Notifications
<div> <div>Patients_chance_to_have_KidneyDisease_with_Albumin</div> <div>text</div> <div>1</div> <div>Zonnya Ab</div> <div>2</div> <div>Skipple Marriot</div> <div>3</div> <div>Mike Bing</div> <div>4</div> <div>Sarah West</div> </div>		
Total rows: 6 of 6 Query complete 00:00:00.034 ✓ Successfully run. Total query runtime: 34 msec. 6 rows affected. ✕		

69. Get the patient's name who has a max speed.

Query:

```
select concat(pt."Firstname" || ' ' || pt."Lastname") from "Patients" pt
join "Walking_Test" wt on wt."WalkTest_ID"=pt."WalkTest_ID"
```


where wt."Gait_DT_Speed" =(select max("Gait_DT_Speed") from "Walking_Test")

45

Data Output Messages Notifications

	concat text
1	Ted Hue

Total rows: 1 of 1 Query complete 00:00:00.041

✓ Successfully run. Total query runtime: 41 msec. 1 rows affected.

70. Write a query to find out the percentage of Lab visits by Lab names.

Query:

Select "Lab_names", to_char(count(*) * 100.0 / sum(count(*)) over(), 'fm99D00%') Percentage_Labvisit
from "Lab_Visit"
group by "Lab_names";

Data Output Messages Notifications

	Lab_names text	percentage_labvisit text
1	Cultivate Lab	16.88%
2	Stream Diagnostix	16.88%
3	The Brain Institute	16.88%
4	Optimal Diagnostics	16.88%
5	Genesis Lab	16.88%
6	GreenDots lab	15.58%

Total rows: 6 of 6 Query complete 00:00:00.037

✓ Successfully run. Total query runtime: 37 msec. 6 rows affected. ✕

71. Write a query to get Patient IDs for verbally cognitively impaired who satisfy any 2 conditions.

--(HINT: dementia/cognitive impaired: any patient who has any two abnormal test results).

Query:

select pt."Patient_ID" VerballyCognitivelyImpaired_2conditions
from "Patients" pt
join "Link_Reference" lr on pt."Link_Reference_ID"=lr."Link_Reference_ID"
join "Verbal_Cognitive" vg on vg."VC_ID"=lr."VC_ID"
where (vg."DS"<13 and "WTAR"<=20)
or ("HVLT"<14 and "WTAR"<=20)

or (vg."VF"<42 and "WTAR"<=20)

Data Output		Messages	Notifications
verballycognitivelyimpaired_2conditions			
1	S0539		
2	S0543		
3	S0550		
4	S0582		

Total rows: 4 of 4 Query complete 00:00:00.043

✓ Successfully run. Total query runtime: 43 msec. 4 rows affected.

72.Display a list of patients who are memory cognitively impaired with the GDS test and whose diabetes duration is between 5 to 30.

Query:

```
SELECT CONCAT (a."Firstname" || ' ' || a."Lastname")
"Patients_MC_GDStest_Diabilities_Duration_Btw_5_and_30"
From "Patients" a
inner join "Link_Reference"b
on a."Link_Reference_ID" = b."Link_Reference_ID"
inner join "Memory_Cognitive" c
on b."MC_ID" = c."MC_ID"
WHERE c. "GDS">=15 and a."Diabetes_Duration" between 5 and 30;
```

Data Output		Messages	Notifications
Patients_MC_GDStest_Diabilities_Duration_Btw_5_and_30			
1	Jerrilyn Kane		
2	Bailey King		
3	Justin Machon		
4	Dwight Steuhlmeier		

Total rows: 6 of 6 Query complete 00:00:00.036

✓ Successfully run. Total query runtime: 36 msec. 6 rows affected. ✕

73. Write a query to the get number of Patient_IDs who visited between March 2005 and March 2006

Query:

```
select count("Patient_ID")Patients_visited_btw_mar05_mar06 from public."Patients"
WHERE "Visit_Date" BETWEEN '2005-03-01' AND '2006-03-31';
```

Data Output Messages Notifications		
<div> <div>patients_visited_btw_mar05_mar06</div> <div>bigint</div> </div>		
1		38
<div>Total rows: 1 of 1 Query complete 00:00:00.035</div> <div> <div>✓ Successfully run. Total query runtime: 35 msec. 1 rows affected.</div> <div>✗</div> </div>		

74. Get the number of patients who visited each lab using the windows function.

Query:

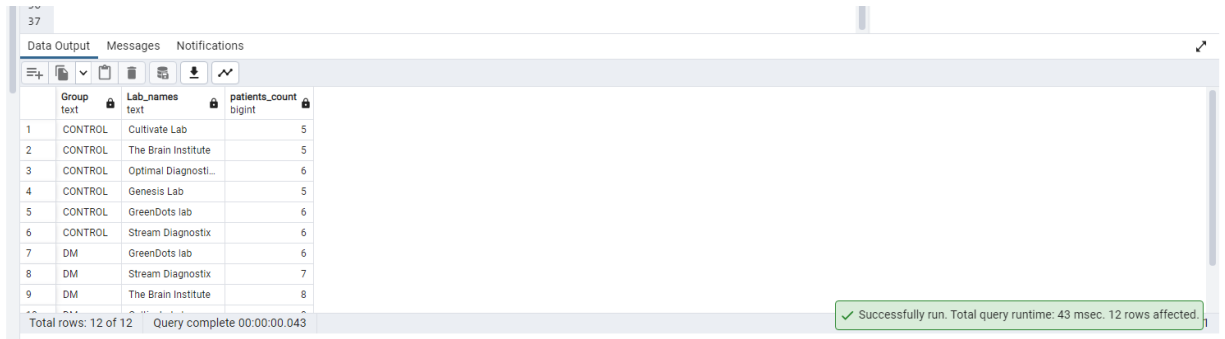
```
SELECT Distinct c."Lab_names",count(*)
over (PARTITION BY c."Lab_names")
FROM "Lab_Visit" c
INNER JOIN "Link_Reference" b ON c."Lab_visit_ID" = b."Lab_visit_ID"
INNER JOIN "Patients" a ON a."Link_Reference_ID" = b."Link_Reference_ID"
group by c."Lab_visit_ID"
```

Data Output Messages Notifications		
<div> <div>Lab_names</div> <div>text</div> </div>		
<div> <div>count</div> <div>bigint</div> </div>		
1	Cultivate Lab	13
2	GreenDots lab	12
3	Optimal Diagnost...	13
4	Genesis Lab	13
5	The Brain Institute	13
6	Stream Diagnostix	13
<div>Total rows: 6 of 6 Query complete 00:00:00.038</div> <div> <div>✓ Successfully run. Total query runtime: 38 msec. 6 rows affected.</div> <div>✗</div> </div>		

75.Find the number of control and DM patients who visited each lab.

Query:

```
select gp."Group",lv."Lab_names", count(*) Patients_Count from "Patients" pt
join "Group" gp on pt."Group_ID "=gp."Group_ID"
join "Link_Reference" lr on lr."Link_Reference_ID "=pt."Link_Reference_ID"
join "Lab_Visit" lv on lv."Lab_visit_ID "=lr."Lab_visit_ID"
group by gp."Group",lv."Lab_names" order by gp."Group"
```



	Group	Lab_names	patients_count
	text	text	bigint
1	CONTROL	Cultivate Lab	5
2	CONTROL	The Brain Institute	5
3	CONTROL	Optimal Diagnosti...	6
4	CONTROL	Genesis Lab	5
5	CONTROL	GreenDots lab	6
6	CONTROL	Stream Diagnostix	6
7	DM	GreenDots lab	6
8	DM	Stream Diagnostix	7
9	DM	The Brain Institute	8
10	DM	Optimal Diagnosti...	6
11	DM	Genesis Lab	5
12	DM	Cultivate Lab	5

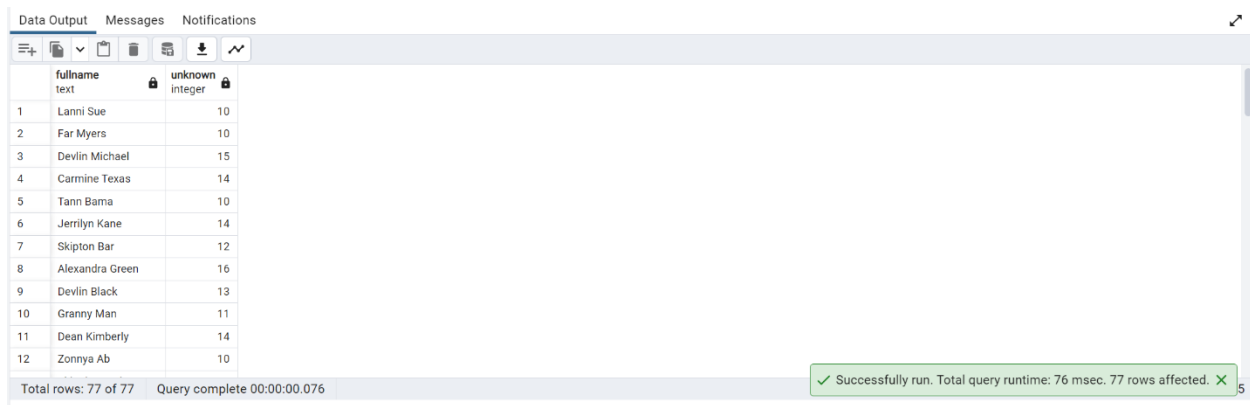
Total rows: 12 of 12 Query complete 00:00:00.043

✓ Successfully run. Total query runtime: 43 msec. 12 rows affected.

76.Please go through the below screenshot and create the exact output.

Query:

```
select concat("Patients"."Firstname" || ' ' || "Patients"."Lastname")as "fullname",
length(concat("Patients"."Firstname" || ' ' || "Patients"."Lastname"))+1 as "unknown" from
public."Patients"
```



	fullname	unknown
	text	integer
1	Lanni Sue	10
2	Far Myers	10
3	Devlin Michael	15
4	Carmin Texas	14
5	Tann Barna	10
6	Jerrilyn Kane	14
7	Skipton Bar	12
8	Alexandra Green	16
9	Devlin Black	13
10	Granny Man	11
11	Dean Kimberly	14
12	Zonnya Ab	10

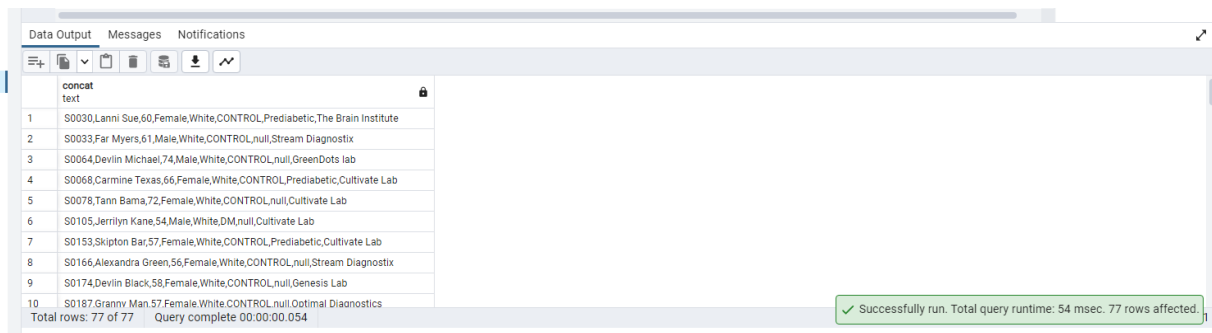
Total rows: 77 of 77 Query complete 00:00:00.076

✓ Successfully run. Total query runtime: 76 msec. 77 rows affected. ✕

77. Write a query to get comma-separated values of patient details .(Use a maximum of 6 columns from different tables)

Query:

```
select concat(pt."Patient_ID" || ',' || pt."Firstname" || ' ' || pt."Lastname" || ',' || pt."Age" || ','
            || gn."Gender" || ',' || r."Race" || ',' || gr."Group" || ',' ||
            (case when gr."Group"='CONTROL' AND lt."Hb_A1C" between 5.7 and 6.4 then
'Prediabetic'
            else 'null'end ) || ',' || lv."Lab_names"
)
from "Patients" pt
join "Gender" gn on pt."Gender_ID"=gn."Gender_ID"
join "Race" r on pt."Race_ID"=r."Race_ID"
join "Group" gr on pt."Group_ID"=gr."Group_ID"
join "Link_Reference" lr on pt."Link_Reference_ID"=lr."Link_Reference_ID"
join "Lab_Test" lt on lt."Lab_ID"=lr."Lab_ID"
join "Lab_Visit" lv on lr."Lab_visit_ID"=lv."Lab_visit_ID"
```



	concat text
1	S0030,Lanni Sue,60,Female,White,CONTROL,Prediabetic,The Brain Institute
2	S0033,Far Myers,61,Male,White,CONTROL,null,Stream Diagnostix
3	S0064,Devin Michael,74,Male,White,CONTROL,null,GreenDots lab
4	S0068,Carmine Texas,66,Female,White,CONTROL,Prediabetic,Cultivate Lab
5	S0078,Tann Bama,72,Female,White,CONTROL,null,Cultivate Lab
6	S0105,Jerrilyn Kane,54,Male,White,DM,null,Cultivate Lab
7	S0153,Skipton Bar,57,Female,White,CONTROL,Prediabetic,Cultivate Lab
8	S0166,Alexandra Green,56,Female,White,CONTROL,null,Stream Diagnostix
9	S0174,Devin Black,58,Female,White,CONTROL,null,Genesis Lab
10	S0187,Granny Man,57,Female,White,CONTROL,null,Optimal Diagnostics

Total rows: 77 of 77 Query complete 00:00:00.054 ✓ Successfully run. Total query runtime: 54 msec. 77 rows affected.

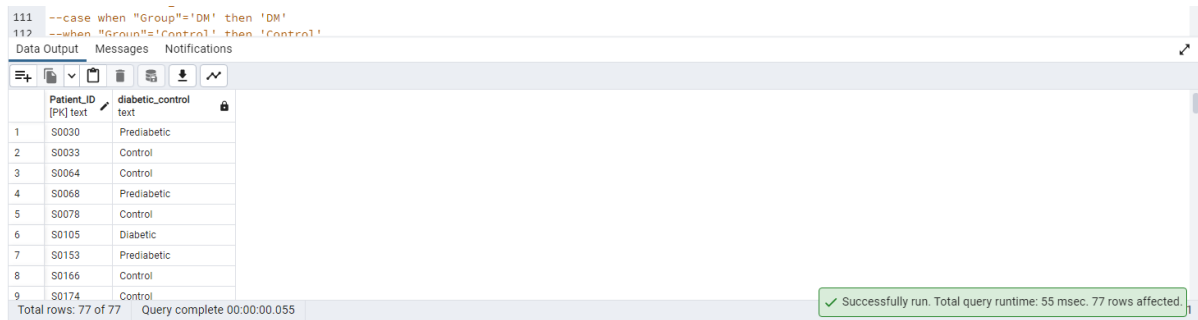
78. Write a query to determine get the Patient IDs ,in DM and Control groups ,that are in prediabetic stage and label them accordingly.

Query:

```

select "Patients"."Patient_ID" ,
case when "Group"='DM' then 'Diabetic'
when "Group"='CONTROL' and "Lab_Test"."Hb_A1C" between 5.7 and 6.4 then 'Prediabetic'
when "Group"='CONTROL' and "Lab_Test"."Hb_A1C" <5.7 then 'Control'
else 'LabResult_Not_Available'
end Diabetic_Control
from "Patients" join "Group" on "Patients"."Group_ID"="Group"."Group_ID"
join "Link_Reference" on "Patients"."Link_Reference_ID"="Link_Reference"."Link_Reference_ID"
join "Lab_Test" on "Lab_Test"."Lab_ID"="Link_Reference"."Lab_ID"

```



Patient_ID	diabetetic_control
1	Prediabetic
2	Control
3	Control
4	Prediabetic
5	Control
6	Diabetic
7	Prediabetic
8	Control
9	Control

Total rows: 77 of 77 Query complete 00:00:00.055 Successfully run. Total query runtime: 55 msec. 77 rows affected.

79. Calculate the Patient's Daytime MAP and Nighttime MAP.

Query:

```

select concat("Patients"."Firstname" || ' ' || "Patients"."Lastname") Patients_Name,
round ((Bp."24Hr_Day_DBP"+((Bp."24Hr_Day_SBP"-Bp."24Hr_Day_DBP")/3))) Daytime_MAP,
round ((Bp."24Hr_Night_DBP"+((Bp."24Hr_Night_SBP"-Bp."24Hr_Night_DBP")/3))) Nighttime_MAP
from "Patients" join "Blood_Pressure" Bp
on "Patients"."BP_ID" = Bp."BP_ID"

```

85

Data Output Messages Notifications

	patients_name text	daytime_map double precision	nighttime_map double precision
1	Lanni Sue	94	82
2	Far Myers	91	92
3	Devlin Michael	86	75
4	Carmine Texas	80	71
5	Tann Bama	86	82
6	Jerrilyn Kane	86	84
7	Skipton Bar	85	79
8	Alexandra Green	77	74
9	Devlin Black	83	80
10	Granny Man	89	76
11	Dean Kimberly	80	76

Total rows: 77 of 77 Query complete 00:00:00.043

✓ Successfully run. Total query runtime: 43 msec. 77 rows affected. Ln 80, Col 1

80. Write a query using recursive view

Query:

```
CREATE RECURSIVE VIEW Age_factor ("Patient_ID", FullName) AS
SELECT
    "Patient_ID",
    concat("Firstname", ' ', "Lastname") as FullName
FROM
    public."Patients"
WHERE
    "Age" = 60
UNION ALL
SELECT
    pt."Patient_ID",
    (
        af.FullName || ' > ' || af.FullName
    ) AS increasing_age
FROM
    public."Patients" pt
INNER JOIN Age_factor af ON pt."Patient_ID"=af."Patient_ID";
```

PIRUTHIHA DHARMAR (She/Her)

Data Output

Messages

Notifications

CREATE VIEW

Query returned successfully in 59 msec.

Total rows: 0 of 0

Query complete 00:00:00.059

✓ Query returned successfully in 59 msec. X

2