**SQL Hackathon January 2024 Extra Queries**

**Team Number: Team1**

**Team Name: Schema Squad**

**Team Members:**

**Total Extra Questions: 7**

**Question 1:**

Display the arrays of the patients, total number of patients for all combinations of hypertension and diabetes comorbidities with having at least one of the two.

**Description:**

Functions used: array\_length, array\_agg

**Query:**

SELECT comorbidity AS comorbidity

,array\_length(ARRAY\_AGG(patient),1)as noOFPatients

,ARRAY\_AGG(patient)as array\_OF\_Patients

FROM(

SELECT patient\_id AS patient

,CASE

WHEN com\_hypertension=1 AND com\_diabetes=1 THEN 'Hypertension-Diabetes'

WHEN com\_diabetes=1 AND com\_hypertension=0 THEN 'Diabetes-noHypertension'

WHEN com\_diabetes=0 AND com\_hypertension=1 THEN 'NoDiabetes-Hypertension'

END AS comorbidity

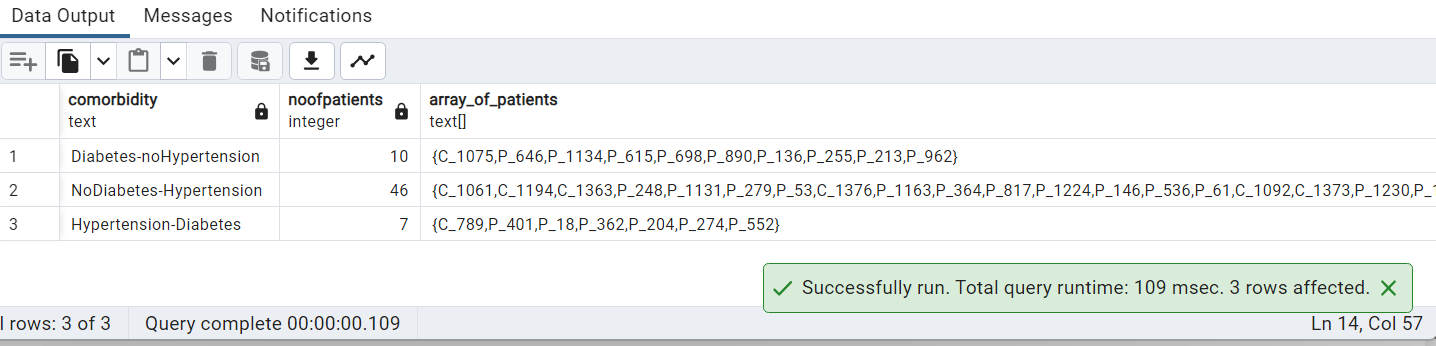
FROM comorbidities

WHERE com\_hypertension=1

OR com\_diabetes=1

) data(patient,comorbidity) GROUP BY comorbidity

**Result:**

****

**Question 2:**

Display all comorbidities having less than 15 patients using unnest function from the result of above question Question 1

**Description:**

Functions used: UNNEST

**Query:**

SELECT UNNEST(d.array\_OF\_Patients) AS patient\_id,d.comorbidity

FROM

(

SELECT comorbidity AS comorbidity

,array\_length(ARRAY\_AGG(patient),1)as noOFPatients

,ARRAY\_AGG(patient)as array\_OF\_Patients

FROM

(

SELECT concat(patient\_id, '') as patient,

CASE

WHEN com\_hypertension=1 AND com\_diabetes=1 THEN 'Hypertension-Diabetes'

WHEN com\_diabetes=1 AND com\_hypertension=0 THEN 'Diabetes-noHypertension'

WHEN com\_diabetes=0 AND com\_hypertension=1 THEN 'NoDiabetes-Hypertension'

END AS comorbidity

FROM comorbidities

WHERE com\_hypertension=1

OR com\_diabetes=1

) AS data(patient,comorbidity)

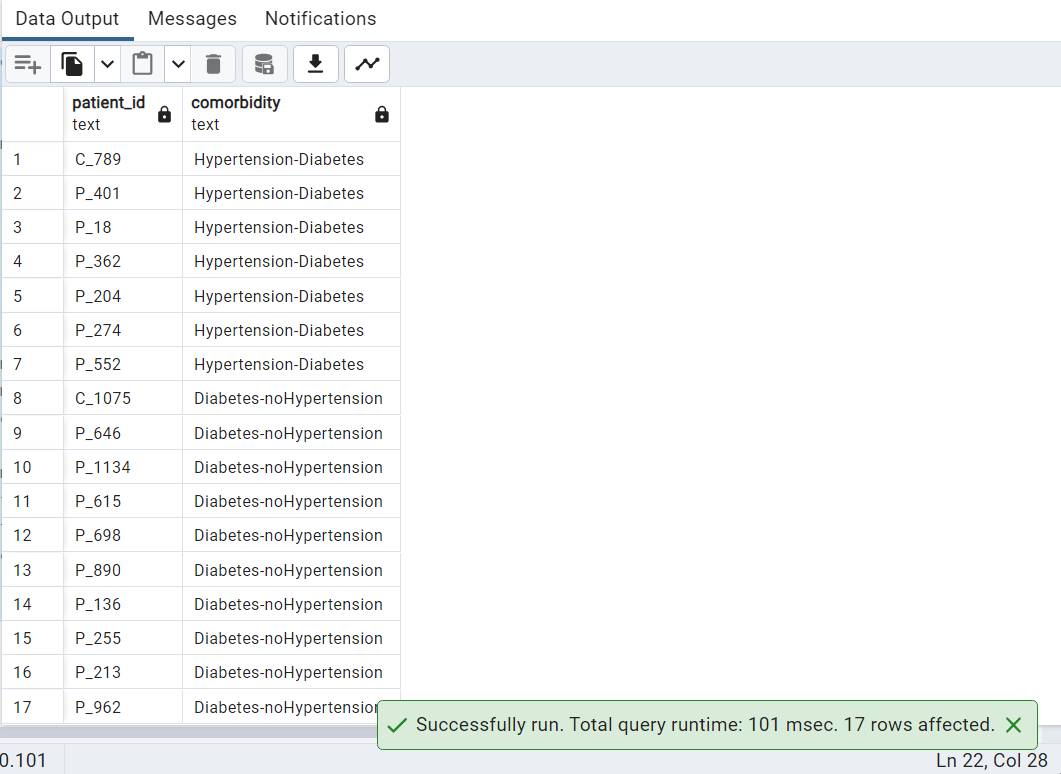
GROUP BY comorbidity

)d

WHERE d.noOfpatients<15

ORDER BY d.noOfpatients

**Result:**

****

**Question 3:**

Find the list of prescribed drug starting with vowels and ends with alphabet 'b' using REGEXP\_LIKE

**Description:**

Functions used: REGEXP\_LIKE

**Query:**

SELECT

DISTINCT(prescribed\_drug) AS drug\_name\_startswith\_vowel\_endswith\_b

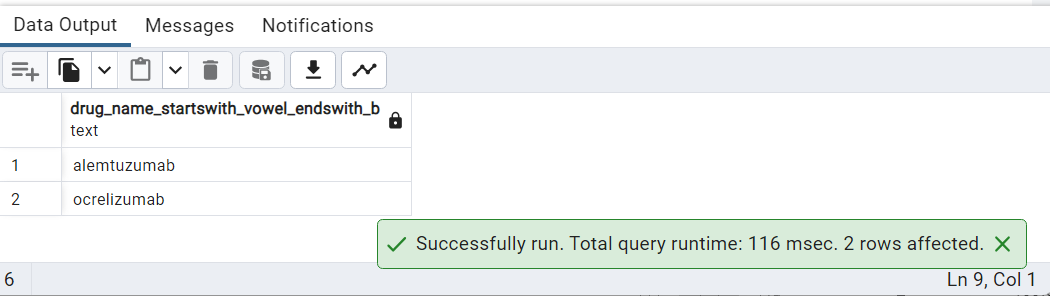
FROM

Patient\_msdetails

WHERE REGEXP\_LIKE(prescribed\_drug,'^[aeiou][\w]\*[b]$','i')

ORDER BY prescribed\_drug;

**Result:**

****

**Question 4:**

Display all the patients self isolated in the last quarter of every year present in dataset along with year and quarter. (4th quarter— from October-December)

**Description:**

Functions used: EXTRACT QUARTER

**Query:**

SELECT

COUNT(patient\_id),

to\_char(date\_trunc('quarter', date\_of\_onset)::date, 'yyyy-q')

FROM

covid\_details

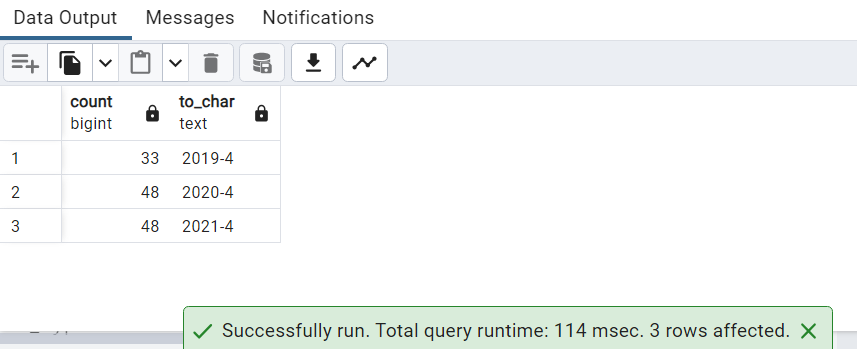
WHERE EXTRACT(QUARTER FROM date\_of\_onset)=4

AND covid19\_self\_isolation=1

GROUP BY to\_char(date\_trunc('quarter', date\_of\_onset)::date, 'yyyy-q')

ORDER BY to\_char(date\_trunc('quarter', date\_of\_onset)::date, 'yyyy-q');

**Result:**

****

**Question 5:**

Display the string of all female pregnant patients,report source in age cat 1 for all ms\_type2

**Description:**

Functions used: STRING\_AGG

**Query:**

SELECT

ms\_type2, string\_agg(patient\_id,',') AS patients

,STRING\_AGG(report\_source,',') AS report\_source

FROM patient\_details

WHERE age\_in\_cat =1

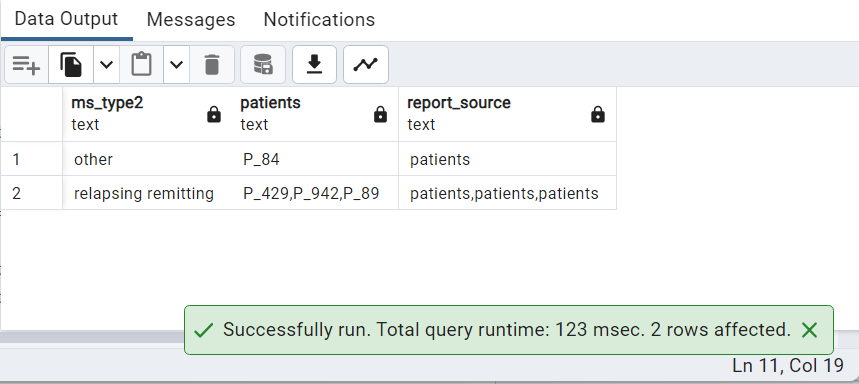
AND sex='female'

AND pregnancy=1

GROUP BY ms\_type2

ORDER BY ms\_type2;

**Result:**

****

**Question 6:**

Write a query displays count of hospitalized patients along with count of patient those are admitted in icu or in self isolation

**Description:**

Functions used: FILTER

**Query:**

SELECT

COUNT(patient\_id),

COUNT(patient\_id) FILTER(WHERE covid19\_self\_isolation=1) iso\_count,

COUNT(patient\_id) FILTER(WHERE covid19\_icu\_stay=1) icu\_count

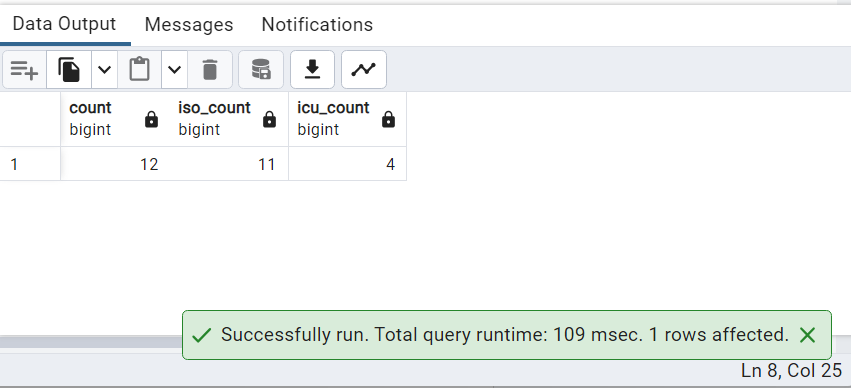
FROM covid\_details

WHERE covid19\_admission\_hospital=1

AND (covid19\_self\_isolation=1

OR covid19\_icu\_stay =1);

**Result:**

****

**Question 7:**

Create a table with JSON data Type. Insert data from patient\_details. Display all female patients data

**Description:**

Functions used: jsonb\_build\_object

**Query:**

CREATE TABLE

covid\_patient\_json\_tbl

(

id serial NOT NULL PRIMARY KEY,

info json NOT NULL

);

INSERT INTO

covid\_patient\_json\_tbl(info)

SELECT

row\_to\_json(x)::jsonb FROM

(

SELECT jsonb\_build\_object('patient\_id', patient\_id) AS patient,

jsonb\_build\_object('sex', sex, 'age\_in\_cat', age\_in\_cat) AS patient\_det

FROM patient\_details

) x;

--DISPLAY ALL PATIENTS DATA

SELECT info

FROM

covid\_patient\_json\_tbl;

--DISPLAYS ALL FEMALE PATIENTS DATA

SELECT info->>'patient' AS patient,

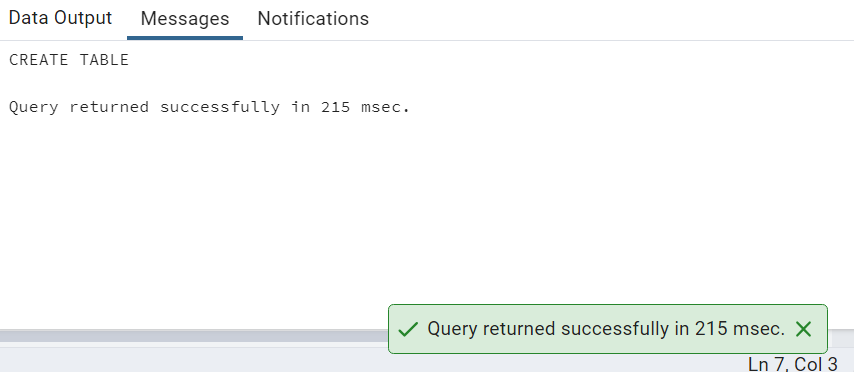
info->>'patient\_det' AS age\_type

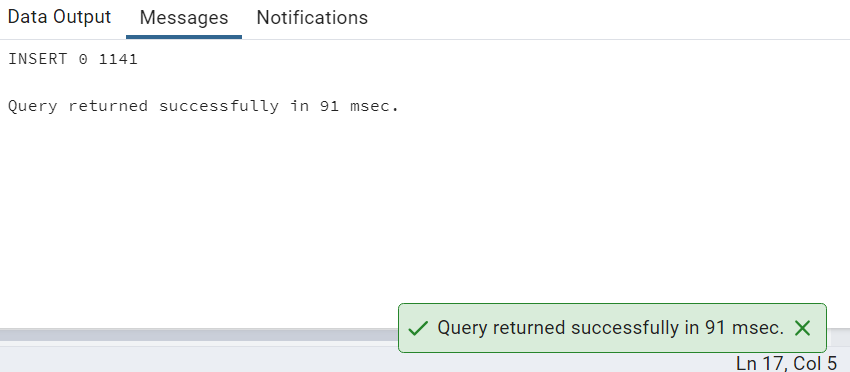
FROM

covid\_patient\_json\_tbl

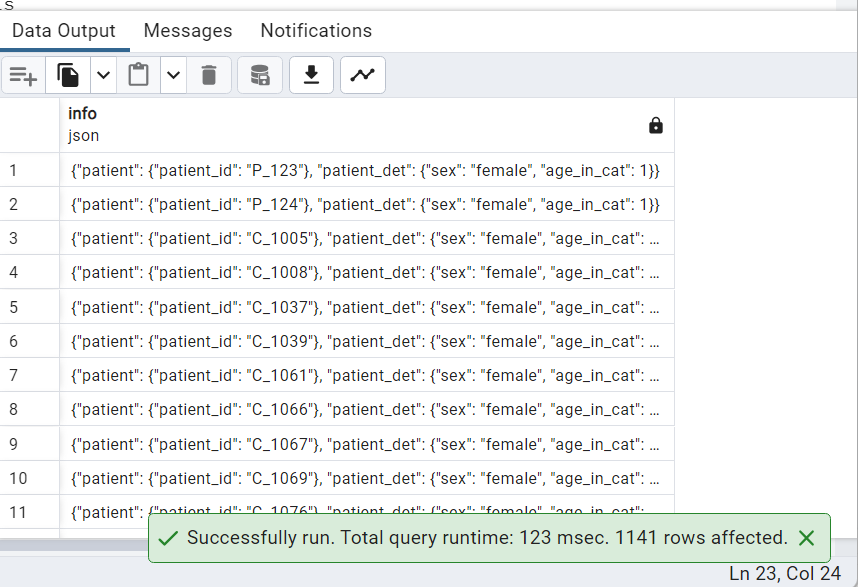
WHERE info ->'patient\_det'->>'sex' ='female'

**Result:**

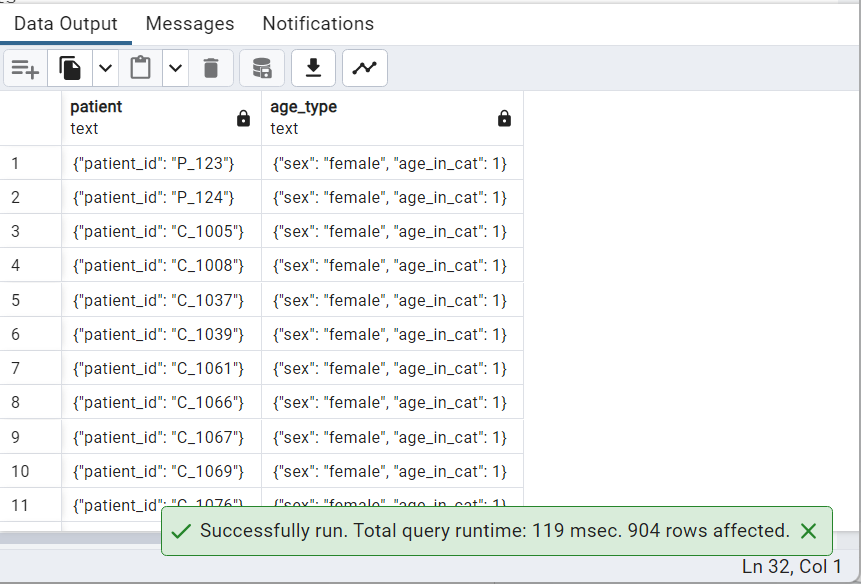
****

****

**LIST OF ALL PATIENTS**

****

**LIST OF ALL FEMALE PATIENTS**

****