

Queries and their output

TOP THREE QUERIES

Q1. Constituency wise candidate rank and margin of loss of candidates for an election

SELECT

```
    CO.Constituency_ID,
    CO.Constituency_Name,
    CD.Candidate_ID,
    V.Name AS Candidate_Name,
    CD.No_of_Votes,
    CD.Party_ID,
    P.Party_Name,
    (
        SELECT COUNT(*)
        FROM Candidate CD2
        WHERE
            CD2.Nomination_Constituency = CD.Nomination_Constituency
            AND CD2.Election_ID          = 'LS2024'
            AND CD2.Nomination_Status    = 'Accepted'
            AND CD2.No_of_Votes          > CD.No_of_Votes
    ) + 1 AS Rank,
    (
        (SELECT MAX(CD3.No_of_Votes)
         FROM Candidate CD3
         WHERE
             CD3.Nomination_Constituency = CD.Nomination_Constituency
             AND CD3.Election_ID          = 'LS2024'
             AND CD3.Nomination_Status    = 'Accepted'
        )
        - CD.No_of_Votes
    ) AS Margin_of_Loss
FROM (SELECT * FROM Candidate WHERE Nomination_Status = 'Accepted' AND
Election_ID = 'LS2024') AS CD
JOIN Constituency CO ON CD.Nomination_Constituency = CO.Constituency_ID
JOIN Voters V ON CD.Voter_ID = V.Voter_ID
LEFT JOIN Parties P ON CD.Party_ID = P.Party_ID
ORDER BY CO.Constituency_Name, Rank;
```

The screenshot shows the pgAdmin 4 interface with a SQL query executed. The query is a complex SELECT statement that calculates the rank and margin of loss for candidates across different constituencies. The results are displayed in a table with 9 rows and 9 columns.

constituency_id	constituency_name	candidate_id	candidate_name	no_of_votes	party_id	party_name	rank	margin_of_loss
GJ001	Ahmedabad East	2024000009	Priya Singh	982378	P002	Indian National Congress	1	0
GJ001	Ahmedabad East	2024000007	Meera Patel	848489	P005	Shiv Sena (UBT)	2	133889
GJ002	Ahmedabad West	2024000011	Harish Mehta	1000099	P004	Trinamool Congress	1	0
GJ002	Ahmedabad West	2024000010	Kiran Shah	724747	P006	Bahujan Samaj Party	2	275352
MH003	Amravati	2024000040	Pratiksha Sawant	1928328	P001	Bharatiya Janata Party	1	0
MH003	Amravati	2024000041	Shubham Yadav	46244	P002	Indian National Congress	2	1882084
RJ003	Banswara	2024000025	Deepak Kumar	0	P008	Nationalist Congress Party	1	0
UP004	Bulandshahr	2024000062	Ravi Singh	789336	P006	Bahujan Samaj Party	1	0
UP004	Bulandshahr	2024000064	Madhavi Reddy	743873	P003	Aam Aadmi Party	2	45463

Q2. Party Wise Vote share earned for a particular election

SELECT

```
P.Party_ID,
P.Party_Name,
SUM(C.No_of_Votes) AS Party_Total_Votes,
(SELECT SUM(C1.No_of_Votes)
FROM Candidate C1
WHERE C1.Nomination_Status = 'Accepted'
AND C1.Election_ID = 'LS2024') AS Total_Votes,
ROUND(
(SUM(C.No_of_Votes) * 100.0) /
(SELECT SUM(C1.No_of_Votes)
FROM Candidate C1
WHERE C1.Nomination_Status = 'Accepted'
AND C1.Election_ID = 'LS2024'),
2
) AS Vote_Share_Percentage
```

FROM

```
(Select * FROM Candidate WHERE Nomination_Status = 'Accepted' AND Election_ID
= 'LS2024') AS C
LEFT JOIN Parties P ON C.Party_ID = P.Party_ID
GROUP BY P.Party_ID;
```

pgAdmin 4

FileObjectToolsEdit ViewWindow Help

Object Explorer

Materialized Views

Operators

Sequences

Tables (22)

booth_election

booth_level_officer

candidate

complaints

constituency

constituency_level_off

deployed_at

election

election_phase

election_schedule

employed_at

employee

evm_vvpatt

national_level_officer

parties

polling_booth

postal_voting

state

state_level_officer

support_staff

vote

voters

Trigger Functions

Types

Views

public

salesmanagement

Subscriptions

public

Login/Group Roles

202301475/202301475@PgSQLServer

No limit

Query

Query History

--Partywise Vote share
SELECT
P.Party_ID,
P.Party_Name,
SUM(C.No.of_Votes) AS Party_Total_Votes,
(SELECT SUM(C1.No.of_Votes)
FROM Candidate C1
WHERE C1.Nomination_Status = 'Accepted'
AND C1.Election_ID = 'LS2024') AS Total_Votes,
ROUND(
(SUM(C.No.of_Votes) * 100.0) /
(SELECT SUM(C1.No.of_Votes)
FROM Candidate C1
WHERE C1.Nomination_Status = 'Accepted'
AND C1.Election_ID = 'LS2024'),
2
) AS Vote_Share_Percentage
FROM

Data OutputMessagesNotifications

Showing rows: 1 to 9Page No: 1 of 1

	party_id [PK] character (10)	party_name character varying (100)	party_total_votes bigint	total_votes bigint	vote_share_percentage numeric
1	[null]	[null]	632183	27022762	2.34
2	P001	Bharatiya Janata Party	6900460	27022762	22.21
3	P007	Communist Party of India (Marxist)	266379	27022762	0.99
4	P003	Aam Aadmi Party	2649756	27022762	9.81
5	P008	Nationalist Congress Party	0	27022762	0.00
6	P006	Bahujan Samaj Party	4906127	27022762	18.16
7	P002	Indian National Congress	4849207	27022762	17.94
8	P005	Shiv Sena (UBT)	4996920	27022762	18.49
9	P004	Tirunamool Congress	2721730	27022762	10.07

Total rows: 9Query complete 00:00:15.9

CRLF Ln 46, Col 45

Q3. Voter turnout during a particular election per constituency

SELECT

```

CO.Constituency_ID,
CO.Constituency_Name,
COUNT(DISTINCT V.Voter_ID) AS Total_Active_Voters,
COUNT(DISTINCT VT.Voter_ID) AS Voters_Turned_Out,
ROUND(
    (COUNT(DISTINCT VT.Voter_ID) * 100.0) /
    COUNT(DISTINCT V.Voter_ID),
    2
) AS Turnout_Percentage

```

FROM Constituency CO

JOIN Polling_Booth PB ON CO.Constituency_ID = PB.Constituency_ID

JOIN

```

(Select Voter_ID, Polling_Booth_ID FROM Voters WHERE Voter_Status='Active') AS
V ON PB.Polling_Booth_ID = V.Polling_Booth_ID

```

LEFT JOIN

```

(Select Voter_ID FROM Vote WHERE Election_ID = 'LS2024' AND Status = 'Voted')
AS VT ON V.Voter_ID = VT.Voter_ID

```

GROUP BY CO.Constituency_ID, CO.Constituency_Name

ORDER BY Turnout_Percentage DESC;

The screenshot shows the pgAdmin 4 interface with a SQL query executed. The query calculates voter turnout percentage per constituency. The results table is as follows:

constituency_id	constituency_name	total_active_voters	voters_turned_out	turnout_percentage
1	GJ004	3	3	100.00
2	RJ003	3	3	100.00
3	UP004	3	2	66.67
4	GJ002	3	2	66.67
5	MH003	3	2	66.67
6	RJ001	3	2	66.67
7	TN001	3	2	66.67
8	TN003	3	2	66.67
9	UP002	3	2	66.67

OTHER QUERIES

Q4. Party Wise number of seats won

SELECT

P.Party_Name,
COUNT(Candidate_ID) AS Seats_Won

FROM

(Select Candidate_ID, Party_ID FROM Candidate C
WHERE Nomination_Status='Accepted' AND Election_ID='LS2024'
AND No_of_Votes = (Select MAX(No_of_Votes) FROM Candidate WHERE
Nomination_Status='Accepted' AND Election_ID='LS2024' AND
Nomination_Constituency=C.Nomination_Constituency)
) AS CD

FULL OUTER JOIN Parties P ON CD.Party_ID = P.Party_ID

GROUP BY P.Party_Name

ORDER BY Seats_Won DESC;

The screenshot shows the pgAdmin 4 interface with a SQL query executed. The query is as follows:

```

1  -- Party wise number of seats earned
2
3  SELECT
4      P.Party_Name,
5      COUNT(Candidate_ID) AS Seats_Won
6  FROM
7      (Select Candidate_ID,Party_ID FROM Candidate C
8       WHERE Nomination_Status='Accepted' AND Election_ID='LS2024'
9       AND No_of_Votes = (Select MAX(No_of_Votes) FROM Candidate WHERE Nomination_Status='Accepted' AND Election_ID='LS2024' AND Nomination_Status='Accepted') AS CD
10     FULL OUTER JOIN
11     Parties P ON CD.Party_ID = P.Party_ID
12  GROUP BY
13      P.Party_Name
14  ORDER BY
15      Seats_Won DESC;
16

```

The results are displayed in a table with the following data:

party_name	seats_won
Indian National Congress	5
Bahujan Samaj Party	4
Bharatiya Janata Party	4
Trinamool Congress	2
[null]	1
Nationalist Congress Party	1
Shiv Sena (UBT)	1
Aam Aadmi Party	1
Communist Party of India (Marxist)	0

Total rows: 9 Query complete 00:00:00.179

S

Q5. Constituency wise details of the candidate won for a particular election

SELECT

```

CS.Constituency_ID,
CD.Candidate_ID,
VT.Name,
CD.Party_ID,
P.Party_Name,
CD.No_of_Votes

```

FROM

```

(Select * FROM Candidate C
 WHERE Nomination_Status='Accepted' AND Election_ID='LS2024'
 AND No_of_Votes = (Select MAX(No_of_Votes) FROM Candidate WHERE
Nomination_Status='Accepted' AND Election_ID='LS2024' AND
Nomination_Constituency=C.Nomination_Constituency)
 ) AS CD
LEFT JOIN Parties P ON P.Party_ID=CD.Party_ID
JOIN Voters VT ON VT.Voter_ID = CD.Voter_ID
RIGHT JOIN Constituency CS ON CS.Constituency_ID = CD.Nomination_Constituency;

```

The screenshot shows the pgAdmin 4 interface with a SQL query executed. The query is as follows:

```

14 P.Party_Name
15 ORDER BY
16 Seats_Won DESC;
17
18 -- Candidates WON
19 SELECT
20 CS.Constituency_ID,
21 CD.Candidate_ID,
22 VI.Name,
23 CD.Party_ID,
24 P.Party_Name,
25 CD.No_of_Votes
26 FROM
27 (Select * FROM Candidate C
28 WHERE Nomination_Status='Accepted' AND Election_ID='LS2024'
29 AND No_of_Votes = (Select MAX(No_of_Votes) FROM Candidate WHERE Nomination_Status='Accepted' AND Election_ID='LS2024' AND Nomination_Status='Accepted') AS CD
30 ) AS CD
31 LEFT JOIN

```

The results table shows the following data:

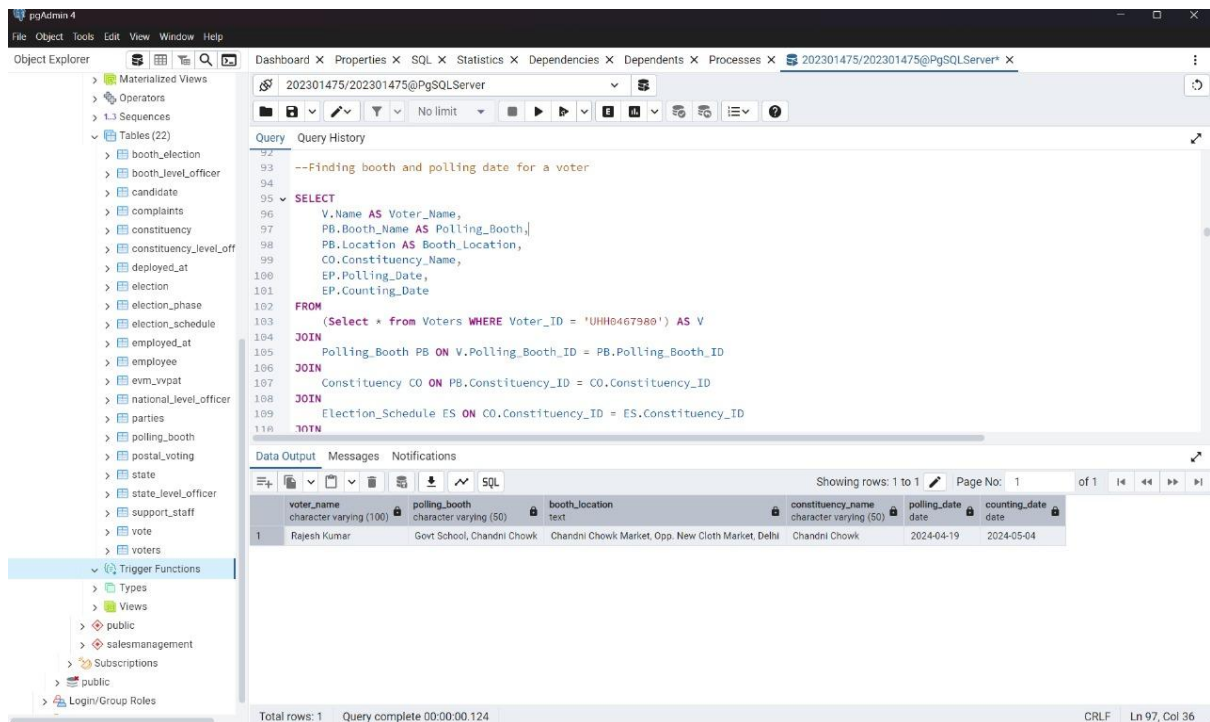
constituency_id	candidate_id	name	party_id	party_name	no_of_votes
DL001	2024000003	Amit Verma	[null]	[null]	500000
DL002	2024000006	Rahul Sharma	P002	Indian National Congress	323234
GJ001	2024000009	Priya Singh	P002	Indian National Congress	982378
GJ002	2024000011	Harish Mehta	P004	Trinamool Congress	1000099
GJ003	[null]	[null]	[null]	[null]	[null]
GJ004	[null]	[null]	[null]	[null]	[null]
LA001	2024000032	Lobsang Tenzin	P004	Trinamool Congress	983093
MH001	2024000035	Mangesh Joshi	P002	Indian National Congress	532635
MH002	2024000038	Avinash Patil	P006	Bahujan Samaj Party	1074347

Q6. Finding booth and polling date for a voter for an election

SELECT

V.Name AS Voter_Name,
PB.Booth_Name AS Polling_Booth,
PB.Location AS Booth_Location,
CO.Constituency_Name,
EP.Polling_Date,
EP.Counting_Date

FROM (Select * from Voters WHERE Voter_ID = 'UHH0467980') AS V
JOIN Polling_Booth PB ON V.Polling_Booth_ID = PB.Polling_Booth_ID
JOIN Constituency CO ON PB.Constituency_ID = CO.Constituency_ID
JOIN Election_Schedule ES ON CO.Constituency_ID = ES.Constituency_ID
JOIN Election_Phase EP ON ES.Election_ID = EP.Election_ID AND ES.Phase_No =
EP.Phase_No
WHERE ES.Election_ID = 'LS2024';



Q7. Support Staff employed at which booth, it's location, start/end date, duty type, constituency, state

SELECT

```

SS.Staff_ID,
SS.Name AS Staff_Name,
SS.Contact_No,
SS.Email AS Staff_Email,
EA.Duty_Type,
EA.Duty_Start_Date,
EA.Duty_End_Date,
EA.Allowance,
PB.Polling_Booth_ID,
PB.Booth_Name,
PB.Location AS Booth_Location,
CO.Constituency_ID,
CO.Constituency_Name,
CO.State_Name

```

FROM Support_Staff SS

JOIN Employed_At EA ON SS.Staff_ID = EA.Staff_ID

JOIN Polling_Booth PB ON EA.Booth_ID = PB.Polling_Booth_ID

JOIN Constituency CO ON PB.Constituency_ID = CO.Constituency_ID

WHERE EA.Election_ID = 'LS2024'

ORDER BY SS.Staff_ID, EA.Duty_Start_Date;

The screenshot shows the pgAdmin 4 interface with a SQL query executed. The query selects staff details from the 'support_staff' table, joining it with 'polling_booth' and 'constituency' tables. The results are displayed in a table with 10 columns: staff_id, staff_name, contact_no, staff_email, duty_type, duty_start_date, duty_end_date, allowance, and polling_booth_id. The results show 8 rows of data.

staff_id	staff_name	contact_no	staff_email	duty_type	duty_start_date	duty_end_date	allowance	polling_booth_id
SS00000001	Rajiv Sharma	9876543210	rajiv.sharma@revenue.gov.in	Presiding Officer	2024-04-19	2024-04-19	5000	DL00100001
SS00000002	Meera Das	8765432109	meera.das@health.gov.in	Polling Assistant	2024-04-20	2024-04-20	4500	GL00100001
SS00000003	Anil Patel	7654321098	anil.patel@transport.gov.in	Polling Officer	2024-04-21	2024-04-21	5000	MH00100001
SS00000004	Priya Singh	6543210987	priya.singh@edu.gov.in	Technical Assistant	2024-04-22	2024-04-22	4500	DL00200001
SS00000005	Anil Patel	7654321098	anil.patel@transport.gov.in	Presiding Officer	2024-04-23	2024-04-23	5000	RL00100001
SS00000006	Priya Singh	6543210987	priya.singh@edu.gov.in	Polling Assistant	2024-04-24	2024-04-24	4500	GL00100002
SS00000007	Priya Singh	6543210987	priya.singh@edu.gov.in	Security Personnel	2024-04-25	2024-04-25	3500	TN00100001
SS00000008	Priya Singh	6543210987	priya.singh@edu.gov.in	Security Personnel	2024-04-26	2024-04-26	3500	TN00100001

Q8. Constituency wise number of active male, female and transgender voters

SELECT

```

CO.Constituency_ID,
CO.Constituency_Name,
(
    SELECT COUNT(*)
    FROM Polling_Booth PB
    JOIN Voters V ON PB.Polling_Booth_ID = V.Polling_Booth_ID
    WHERE
        PB.Constituency_ID = CO.Constituency_ID
        AND V.Gender='M' AND V.Voter_Status='Active'
) AS Male_Voters,
(
    SELECT COUNT(*)
    FROM Polling_Booth PB
    JOIN Voters V ON PB.Polling_Booth_ID = V.Polling_Booth_ID
    WHERE
        PB.Constituency_ID = CO.Constituency_ID
        AND V.Gender='F' AND V.Voter_Status='Active'
) AS Female_Voters,
(
    SELECT COUNT(*)
    FROM Polling_Booth PB
    JOIN Voters V ON PB.Polling_Booth_ID = V.Polling_Booth_ID
    WHERE
        PB.Constituency_ID = CO.Constituency_ID
        AND V.Gender='O' AND V.Voter_Status='Active'
) AS Transgender_Voters

```



```
FROM Constituency C0
ORDER BY C0.Constituency_Name;
```

The screenshot shows the pgAdmin 4 interface. On the left is the Object Explorer with a tree view of the database schema. The main pane displays a SQL query in the Query Editor. The query is a multi-part SELECT statement designed to count male, female, and transgender voters by constituency. It uses subqueries and JOINs to achieve this. Below the query editor, the Data Output pane shows the results of the query as a table with 5 columns: constituency_id, constituency_name, male_voters, female_voters, and transgender_voters. The results show data for 9 constituencies. The status bar at the bottom indicates 'Total rows: 22' and 'Query complete 00:00:00.172'.

```

-- Constituency wise number of male, female and transgender voters
SELECT
  C0.Constituency_ID,
  C0.Constituency_Name,

  /* count male voters in this constituency */
  (
    SELECT COUNT(*)
    FROM Polling_Booth PB
    JOIN Voters V ON PB.Polling_Booth_ID = V.Polling_Booth_ID
    WHERE
      PB.Constituency_ID = C0.Constituency_ID
      AND V.Gender='M' AND V.Voter_Status='Active'
  ) AS Male_Voters,

  /* count female voters in this constituency */
  (
    SELECT COUNT(*)
    FROM Polling_Booth PB
    JOIN Voters V ON PB.Polling_Booth_ID = V.Polling_Booth_ID
    WHERE
      PB.Constituency_ID = C0.Constituency_ID
      AND V.Gender='F' AND V.Voter_Status='Active'
  ) AS Female_Voters,

  /* count transgender voters in this constituency */
  (
    SELECT COUNT(*)
    FROM Polling_Booth PB
    JOIN Voters V ON PB.Polling_Booth_ID = V.Polling_Booth_ID
    WHERE
      PB.Constituency_ID = C0.Constituency_ID
      AND V.Gender='T' AND V.Voter_Status='Active'
  ) AS Transgender_Voters
FROM Constituency C0
ORDER BY C0.Constituency_Name;
```

constituency_id	constituency_name	male_voters	female_voters	transgender_voters
GJ001	Ahmedabad East	1	2	0
GJ002	Ahmedabad West	2	1	0
MH003	Amravati	2	1	0
RJ003	Banswara	2	1	0
UP004	Bulandshahr	2	1	0
DL001	Chandni Chowk	3	2	0
TN002	Chennai North	1	2	0
TN003	Chennai South	2	1	0
RJ004	Chittorgarh	1	2	0

Q9. Find the Candidates who are Contesting from Multiple Constituencies During a particular election

```
SELECT * from Candidate
WHERE Election_ID='LS2024' AND Nomination_Status='Accepted' AND
Voter_ID IN (Select Voter_ID FROM Candidate
WHERE Nomination_Status='Accepted' AND Election_ID='LS2024'
GROUP BY Voter_ID
HAVING count(Candidate_ID)>1)
ORDER BY Candidate_ID;
```

pgAdmin 4

Object Explorer

- Materialized Views
- Operators
- Sequences
- Tables (22)
 - booth_election
 - booth_level_officer
 - candidate
 - complaints
 - constituency
 - constituency_level_off
 - deployed_at
 - election
 - election_phase
 - election_schedule
 - employed_at
 - employee
 - evm_vvpat
 - national_level_officer
 - parties
 - polling_booth
 - postal_voting
 - state
 - state_level_officer
 - support_staff
 - vote
 - voters
- Trigger Functions
- Types
- Views
- public
- salesmanagement
- Subscriptions
- public
- Login/Group Roles

Dashboard X Properties X SQL X Statistics X Dependencies X Dependents X Processes X 202301475/202301475@PgSQLServer X

202301475/202301475@PgSQLServer

Query Query History

```

313
314
315 -- Find the Candidates who are Contesting from Multiple Constituencies During a particular election
316
317 SELECT * FROM Candidate
318 WHERE Election_ID='LS2024' AND Nomination_Status='Accepted' AND
319 Voter_ID IN (SELECT Voter_ID FROM Candidate
320 WHERE Nomination_Status='Accepted' AND Election_ID='LS2024'
321 GROUP BY Voter_ID
322 HAVING count(Candidate_ID)>1)
323 ORDER BY Candidate_ID;
324

```

Data Output Messages Notifications

Showing rows: 1 to 2 Page No: 1 of 1

	candidate_id	voter_id	nomination_constituency	party_id	election_id	educational_qualification	liabilities_value	assets_value	criminal_record
1	2024000046	XCV2345678	TN002	P003	LS2024	Graduate	200000	1300000	[null]
2	2024000064	XCV2345678	UP004	P003	LS2024	Graduate	200000	1300000	[null]

Total rows: 2 Query complete 00:00:00.245 CRLF Ln 313, Col 1

Q10. Constituencies which have remained vacant for an election

```

SELECT C.Constituency_ID, C.Constituency_Name, C.State_Name
FROM Constituency C
LEFT JOIN Candidate CD
ON C.Constituency_ID = CD.Nomination_Constituency AND CD.Nomination_Status =
'Accepted' AND CD.Election_ID = 'LS2019'
WHERE CD.Candidate_ID IS NULL;

```

pgAdmin 4

Object Explorer

- Materialized Views
- Operators
- Sequences
- Tables (22)
 - booth_election
 - booth_level_officer
 - candidate
 - complaints
 - constituency
 - constituency_level_off
 - deployed_at
 - election
 - election_phase
 - election_schedule
 - employed_at
 - employee
 - evm_vvpat
 - national_level_officer
 - parties
 - polling_booth
 - postal_voting
 - state
 - state_level_officer
 - support_staff
 - vote
 - voters
- Trigger Functions
- Types
- Views
- public
- salesmanagement
- Subscriptions
- public
- Login/Group Roles

Dashboard X Properties X SQL X Statistics X Dependencies X Dependents X Processes X 202301475/202301475@PgSQLServer X

202301475/202301475@PgSQLServer

Query Query History

```

220 C.Constituency_Name;
221
222 -- Vacant constituencies for an election
223
224 SELECT
225 C.Constituency_ID,
226 C.Constituency_Name,
227 C.State_Name
228 FROM
229 Constituency C
230 LEFT JOIN
231 Candidate CD
232 ON C.Constituency_ID = CD.Nomination_Constituency
233 AND CD.Nomination_Status = 'Accepted'
234 AND CD.Election_ID = 'LS2019'
235 WHERE
236 CD.Candidate_ID IS NULL;
237
238 -- Checking whether support staff has been given overlapping duties

```

Data Output Messages Notifications

Showing rows: 1 to 3 Page No: 1 of 1

	constituency_id	constituency_name	state_name
1	GJ003	Surendranagar	Gujarat
2	GJ004	Valsad	Gujarat
3	RJ001	Ganganagar	Rajasthan

Total rows: 3 Query complete 00:00:00.225 CRLF Ln 227, Col 17

Successfully run. Total query runtime: 225 msec. 3 rows affected.

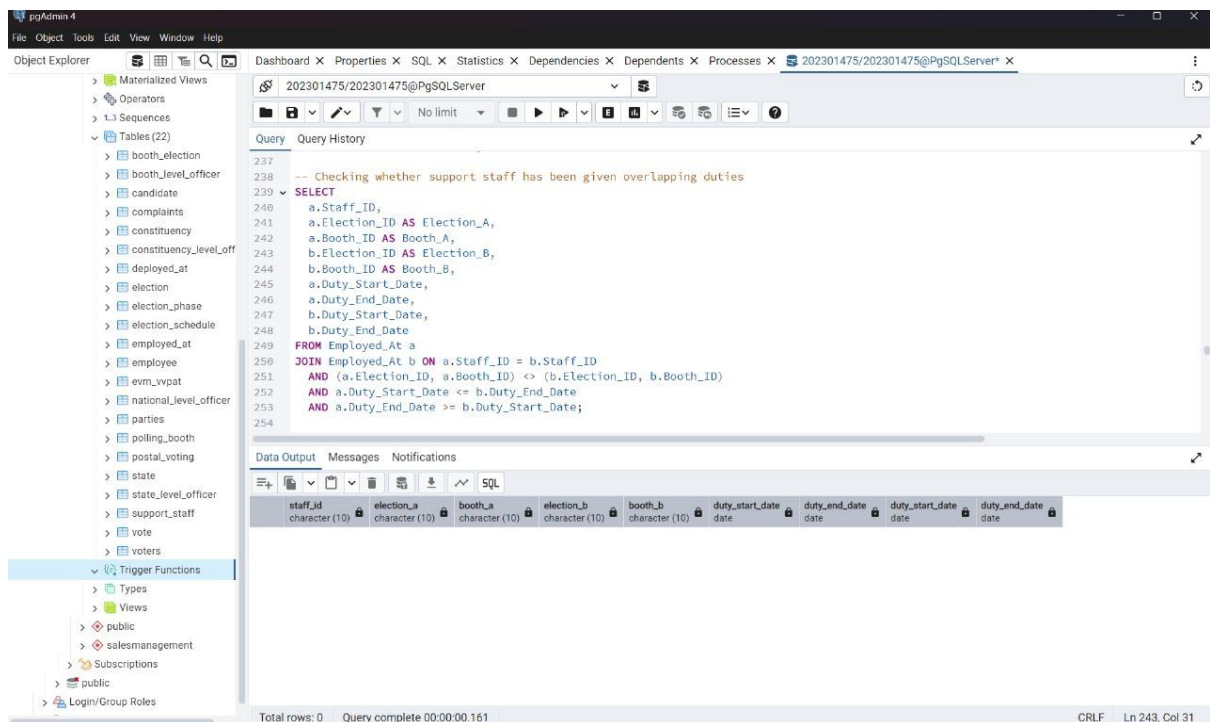
Q11. Checking whether support staff has been given overlapping duties

SELECT

```
a.Staff_ID,  
a.Election_ID AS Election_A,  
a.Booth_ID AS Booth_A,  
b.Election_ID AS Election_B,  
b.Booth_ID AS Booth_B,  
a.Duty_Start_Date,  
a.Duty_End_Date,  
b.Duty_Start_Date,  
b.Duty_End_Date
```

FROM Employed_At a

```
JOIN Employed_At b ON a.Staff_ID = b.Staff_ID  
AND (a.Election_ID, a.Booth_ID) <> (b.Election_ID, b.Booth_ID)  
AND a.Duty_Start_Date <= b.Duty_End_Date  
AND a.Duty_End_Date >= b.Duty_Start_Date;
```



Q12. Getting the officer hierarchy to contact for a voter Booth Level

```
SELECT emp.Officer_ID, vt.name, emp.Designation, emp.Office_Add, emp.  
Office_Contact_No FROM  
(Select Polling_Booth_ID FROM Voters WHERE Voter_ID = 'UHH0467980') b JOIN  
Booth_Level_Officer bl ON bl.Booth_ID = b.Polling_Booth_ID  
JOIN Employee emp ON emp.Officer_ID = bl.Officer_ID  
JOIN Voters vt ON emp.Voter_ID = vt.Voter_ID;
```

The screenshot shows the pgAdmin 4 interface with a SQL query executed. The query is as follows:

```

249 FROM Employed_At a
250 JOIN Employed_At b ON a.Staff_ID = b.Staff_ID
251 AND (a.Election_ID, a.Booth_ID) <> (b.Election_ID, b.Booth_ID)
252 AND a.Duty_Start_Date <= b.Duty_End_Date
253 AND a.Duty_End_Date >= b.Duty_Start_Date;
254
255 -- Getting the officer hierarchy to contact for a voter
256
257 --Booth Level
258
259 SELECT emp.Officer_ID,vt.name, emp.Designation, emp.Office_Add, emp. Office_Contact_No FROM
260 (Select Polling_Booth_ID FROM Voters WHERE Voter_ID = 'UHH0467980') b JOIN Booth_Level_Officer bl ON bl.Booth_ID = b.Polling_Booth_ID
261 JOIN Employee emp ON emp.Officer_ID = bl.Officer_ID
262 JOIN Voters vt ON emp.Voter_ID = vt.Voter_ID;
263

```

The results table shows the following data:

officer_id	name	designation	office_add	office_contact_no
EMP0000032	Anjali Iyer	BLO	Booth 19, Vadodara	9876500027

Total rows: 1 Query complete 00:00:00.150

Constituency Level

```

SELECT emp.Officer_ID,vt.name, emp.Designation, emp.Office_Add, emp.
Office_Contact_No FROM
(Select Polling_Booth_ID FROM Voters WHERE Voter_ID = 'UHH0467980') b
JOIN Polling_Booth pb on b.Polling_Booth_ID = pb.Polling_Booth_ID
JOIN Constituency_Level_Officer cl ON cl.Constituency_ID = pb.Constituency_ID
JOIN Employee emp ON emp.Officer_ID = cl.Officer_ID
JOIN Voters vt ON emp.Voter_ID = vt.Voter_ID;

```

The screenshot shows the pgAdmin 4 interface with a SQL query executed. The query is as follows:

```

280
281
282 -- Election Schedule of a Constituency for a particular Election
283
284 SELECT CS.Constituency_ID, CS.Constituency_Name, EP.*
285 FROM (Select Constituency_ID, Constituency_Name FROM Constituency WHERE Constituency_ID='DL001') AS CS
286 JOIN Election_Schedule ES ON ES.Constituency_ID = CS.Constituency_ID AND ES.Election_ID='LS2024'
287 JOIN Election_Phase EP ON EP.Election_ID = ES.Election_ID AND EP.Phase_NO = ES.Phase_NO;
288

```

The results table shows the following data:

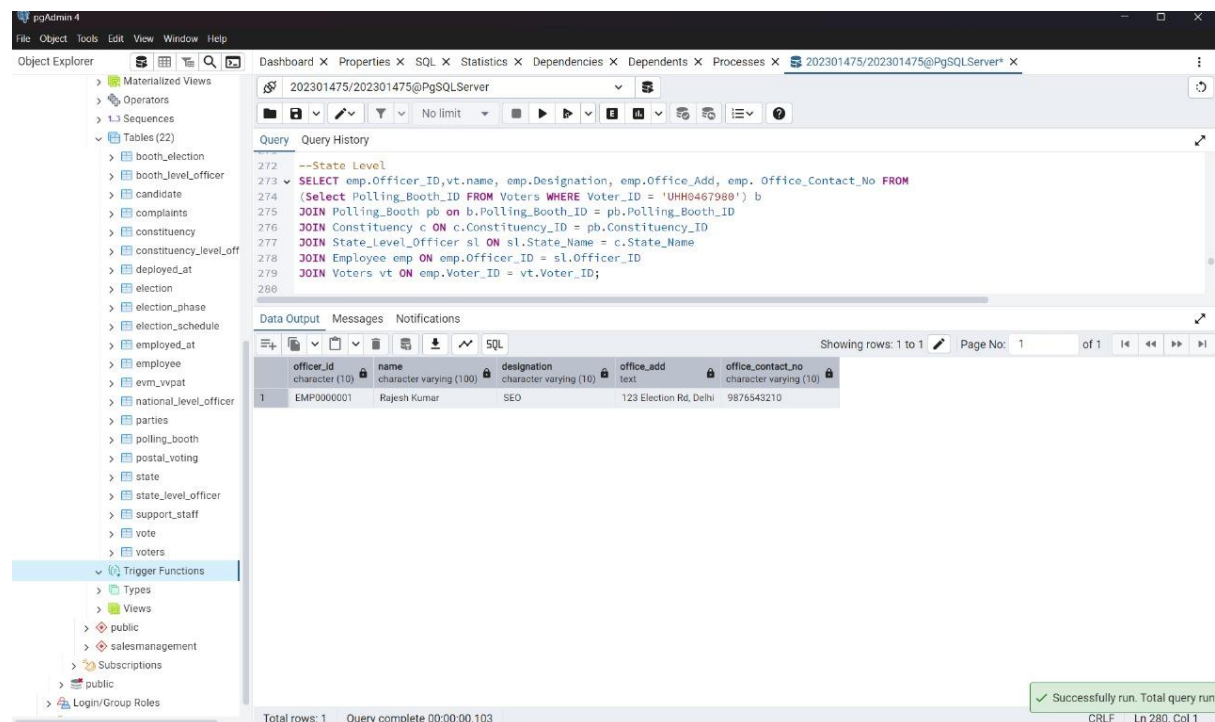
constituency_id	constituency_name	election_id	phase_no	gazette_issue_date	nomination_scrutiny_date	withdrawal_last_date	polling_date	counting_date
DL001	Chandni Chowk	LS2024	1	2024-04-05	2024-04-08	2024-04-12	2024-04-19	2024-05-04

Total rows: 1 Query complete 00:00:00.103

Successfully run. Total query runtime: 103 msec. 1 rows affect

State Level

```
SELECT emp.Officer_ID,vt.name, emp.Designation, emp.Office_Add, emp.
Office_Contact_No FROM
(select Polling_Booth_ID FROM Voters WHERE Voter_ID = 'UHH0467980') b
JOIN Polling_Booth pb on b.Polling_Booth_ID = pb.Polling_Booth_ID
JOIN Constituency c ON c.Constituency_ID = pb.Constituency_ID
JOIN State_Level_Officer sl ON sl.State_Name = c.State_Name
JOIN Employee emp ON emp.Officer_ID = sl.Officer_ID
JOIN Voters vt ON emp.Voter_ID = vt.Voter_ID;
```



Q13. Complete Election Schedule of a Constituency for a particular Election

```
SELECT CS.Constituency_ID, CS.Constituency_Name, EP.*
FROM (select Constituency_ID,Constituency_Name FROM Constituency WHERE
Constituency_ID='DL001') AS CS
JOIN Election_Schedule ES ON ES.Constituency_ID = CS.Constituency_ID AND
ES.Election_ID='LS2024'
JOIN Election_Phase EP ON EP.Election_ID = ES.Election_ID AND EP.Phase_NO =
ES.Phase_NO;
```

Q14. Booth wise number of Complaints

```
SELECT PB.Polling_Booth_ID,PB.Booth_Name,COUNT(Complaint_ID) AS
No_of_Complaints FROM
```



```
(Select * from Complaints WHERE Election_ID='LS2024' ) AS C
RIGHT JOIN Polling_Booth PB ON PB.Polling_Booth_ID = C.Booth_ID
GROUP BY PB.Polling_Booth_ID,PB.Booth_Name;
```

The screenshot shows the pgAdmin 4 interface. On the left is the Object Explorer with a tree view of database objects. The main pane displays a SQL query in the Query editor. Below the query is the Data Output pane showing the results of the query. The query is a right join between the Complaints table (aliased as C) and the Polling_Booth table (aliased as PB) on the basis of Election_ID and Booth_ID. The results are grouped by Polling_Booth_ID and Booth_Name, showing the count of complaints for each booth in the LS2024 election.

polling_booth_id	booth_name	no_of_complaints
1	TN03000002	0
2	TN00100001	1
3	GJ00100001	0
4	DL00100002	0
5	GJ00200001	0
6	UP00200001	1
7	UP00400001	0
8	UP00200002	1
9	LA00100001	0
10	MH00300002	0
11	PB00100002	1
12	RJ00100001	0
13	UP00300001	0
14	TN00200002	0
15	MH00200002	0
16	GJ00100002	0

Q15. All details of Candidates with Criminal Record

```
SELECT CD.Candidate_ID, VT.Name AS Candidate_Name, CD.Criminal_Record,
CD.Nomination_Constituency, CD.Nomination_Status, CD.Election_ID, CD.Party_ID,
P.Party_Name
FROM (Select * FROM Candidate WHERE Criminal_Record IS NOT NULL) AS CD
JOIN Voters VT ON CD.Voter_ID = VT.Voter_ID
LEFT JOIN Parties P ON CD.Party_ID = P.Party_ID;
```


pgAdmin 4

FileObjectToolsEditViewWindowHelp

Object Explorer

Materialized Views

Operators

Sequences

Tables (22)

booth_election

booth_level_officer

candidate

complaints

constituency

constituency_level_off

deployed_at

election

election_phase

election_schedule

employed_at

employee

evm_vvp

national_level_officer

parties

polling_booth

postal_voting

state

state_level_officer

support_staff

vote

voters

Trigger Functions

Types

Views

public

salesmanagement

Subscriptions

public

Login/Group Roles

DashboardPropertiesSQLStatisticsDependenciesDependentsProcesses202301475/202301475@PgSQLServer

202301475/202301475@PgSQLServer

No limit

QueryQuery History

-- All details of Candidates with Criminal Record

SELECT CD.Candidate_ID, VT.Name AS Candidate_Name, CD.Criminal_Record, CD.Nomination_Constituency, CD.Nomination_Status, CD.Election_ID

FROM (Select * FROM Candidate WHERE Criminal_Record IS NOT NULL) AS CD

JOIN Voters VT ON CD.Voter_ID = VT.Voter_ID

LEFT JOIN Parties P ON CD.Party_ID = P.Party_ID;

Data OutputMessagesNotifications

Showing rows: 1 to 37Page No: 1of 1

candidate_id	candidate_name	criminal_record	nomination_constituency	nomination_status	election_id	party_id	party_name
character (10)	character varying (100)	character varying (100)	character (10)	character varying (10)	character (10)	character (10)	character varying (100)
1	2019000001	Rajesh Kumar	Minor Traffic Violation	DL001	Accepted	LS2019	P003
2	2019000003	Amit Verma	Forgery	DL001	Rejected	LS2019	[null]
3	2019000006	Rahul Sharma	Fraud	DL002	Rejected	LS2019	[null]
4	2019000009	Priya Singh	Fraud	GJ001	Accepted	LS2019	[null]
5	2024000011	Harish Mehta	Minor Traffic Violation	GJ002	Accepted	LS2024	P004
6	2019000012	Amit Patel	Forgery	GJ002	Rejected	LS2019	[null]
7	2024000015	Kavita Sen	Fraud	RJ001	Rejected	LS2024	[null]
8	2019000015	Vikram Rathore	Forgery	RJ002	Rejected	LS2019	[null]
9	2024000017	Ajrun Nair	Minor Violation	TN001	Accepted	LS2024	P002
10	2024000018	Lakshmi Iyer	Fraud	TN001	Accepted	LS2024	[null]
11	2019000018	Lakshmi Iyer	Forgery	TN002	Accepted	LS2019	[null]
12	2024000027	Ankit Verma	Fraud	RJ003	Rejected	LS2024	[null]
13	2019000027	Ankit Verma	Forgery	RJ003	Rejected	LS2019	[null]
14	2024000030	Kavita Sen	Minor Violation	RJ004	Rejected	LS2024	[null]
15	2019000030	Kavita Sen	Fraud	RJ004	Rejected	LS2019	[null]
16	2024000033	Tsering Dolma	Minor Violation	LA001	Rejected	LS2024	[null]

Total rows: 37Query complete 00:00:00.120

CRLF

Ln 292, Col 1

Q16. Find the Machines that are not in working condition and deployed

```

Select D.Device_ID, D.Machine_Type,D.Status, D.Booth_ID, B.Booth_Name FROM
(SELECT DA.Device_ID, EV.Machine_Type,EV.Status, DA.Booth_ID
FROM (Select * FROM DEPLOYED_AT WHERE Election_ID='LS2024') AS DA
JOIN EVM_VVPAT AS EV ON EV.Device_ID = DA.Device_ID
WHERE EV.Status <> 'Working') AS D
JOIN Polling_Booth B ON B.Polling_Booth_ID = D.Booth_ID;

```

pgAdmin 4

FileObjectToolsEditViewWindowHelp

Object Explorer

Materialized Views

Operators

Sequences

Tables (22)

booth_election

booth_level_officer

candidate

complaints

constituency

constituency_level_off

deployed_at

election

election_phase

election_schedule

employed_at

employee

evm_vvp

national_level_officer

parties

polling_booth

postal_voting

state

state_level_officer

support_staff

vote

voters

Trigger Functions

Types

Views

public

salesmanagement

Subscriptions

public

Login/Group Roles

DashboardPropertiesSQLStatisticsDependenciesDependentsProcesses202301475/202301475@PgSQLServer

202301475/202301475@PgSQLServer

No limit

QueryQuery History

-- Find the Machines that are not in working condition and deployed

Select D.Device_ID, D.Machine_Type,D.Status, D.Booth_ID, B.Booth_Name FROM
(SELECT DA.Device_ID, EV.Machine_Type,EV.Status, DA.Booth_ID
FROM (Select * FROM DEPLOYED_AT WHERE Election_ID='LS2024') AS DA
JOIN EVM_VVPAT AS EV ON EV.Device_ID = DA.Device_ID
WHERE EV.Status <> 'Working') AS D
JOIN Polling_Booth B ON B.Polling_Booth_ID = D.Booth_ID;

Data OutputMessagesNotifications

Showing rows: 1 to 29Page No: 1of 1

device_id	machine_type	status	booth_id	booth_name	
character (10)	character varying (19)	character varying (20)	character (10)	character varying (50)	
1	VPT0000003	VVPAT	In-Maintenance	DL00200001	Delhi Public School, Rohini
2	EV00000003	EVM	In-Maintenance	DL00200001	Delhi Public School, Rohini
3	VPT0000003	VVPAT	In-Maintenance	GJ00200001	Govt School, Vastrapur
4	EV00000003	EVM	In-Maintenance	GJ00200001	Govt School, Vastrapur
5	VPT0000005	VVPAT	Damaged	GJ00300001	Patidar Community Hall, Ratanpar
6	EV00000005	EVM	Damaged	GJ00300001	Patidar Community Hall, Ratanpar
7	EV00000007	EVM	In-Maintenance	GJ00400001	Govt School, Valsad
8	VPT0000008	VVPAT	In-Maintenance	GJ00400002	Community Hall, Valsad
9	VPT0000003	VVPAT	In-Maintenance	MH00200001	Govt School, Dhule
10	EV00000003	EVM	In-Maintenance	MH00200001	Govt School, Dhule
11	VPT0000005	VVPAT	Damaged	MH00300001	Govt School, Amravati
12	EV00000005	EVM	Damaged	MH00300001	Govt School, Amravati
13	VPT0000003	VVPAT	In-Maintenance	RJ00200001	Govt School, Udaipur
14	EV00000003	EVM	In-Maintenance	RJ00200001	Govt School, Udaipur
15	VPT0000005	VVPAT	Damaged	RJ00300001	Govt School, Banswara
16	EV00000005	EVM	Damaged	RJ00300001	Govt School, Banswara

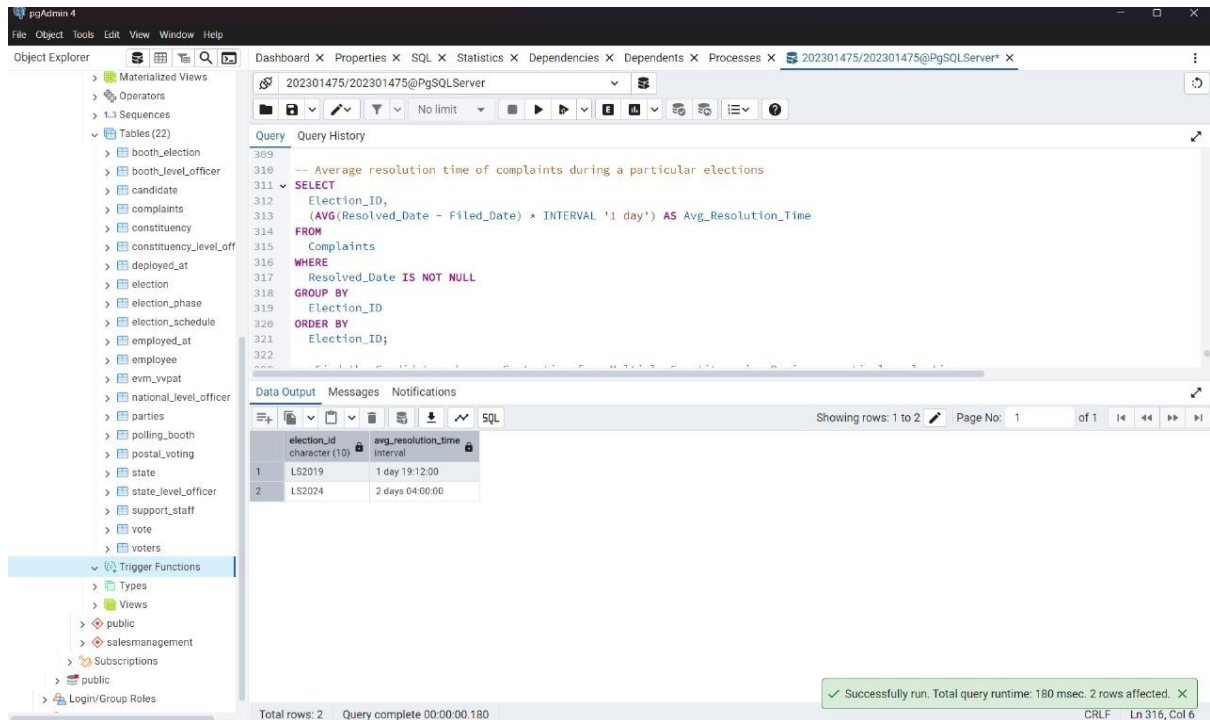
Total rows: 29Query complete 00:00:00.122

Successfully run. Total query runtime: 122 msec. 29 rows affected.

CRLF Ln 308, Col 57

Q17. Average resolution time of complaints during a particular elections

```
SELECT
Election_ID,(AVG(Resolved_Date - Filed_Date) * INTERVAL '1 day') AS
Avg_Resolution_Time
FROM Complaints
WHERE Resolved_Date IS NOT NULL
GROUP BY Election_ID
ORDER BY Election_ID;
```



The screenshot shows the pgAdmin 4 interface with a SQL query executed. The query is as follows:

```
-- Average resolution time of complaints during a particular elections
SELECT
Election_ID,
(AVG(Resolved_Date - Filed_Date) * INTERVAL '1 day') AS Avg_Resolution_Time
FROM
Complaints
WHERE
Resolved_Date IS NOT NULL
GROUP BY
Election_ID
ORDER BY
Election_ID;
```

The results are displayed in a table with the following data:

	election_id character (10)	avg_resolution_time interval
1	LS2019	1 day 19:12:00
2	LS2024	2 days 04:00:00

The status bar at the bottom indicates: Total rows: 2 Query complete 00:00:00.180. A message box at the bottom right states: Successfully run, Total query runtime: 180 msec. 2 rows affected.

--Q18. Pending Complaints Sorted By Filing Date

```
SELECT * FROM Complaints
WHERE Complaint_Status='Pending'
ORDER BY Filed_Date;
```

pgAdmin 4

Object Explorer

- Materialized Views
- Operators
- Sequences
- Tables (22)
 - booth_election
 - booth_level_officer
 - candidate
 - complaints
 - constituency
 - constituency_level_off
 - deployed_at
 - election
 - election_phase
 - election_schedule
 - employed_at
 - employee
 - evm_vvpat
 - national_level_officer
 - parties
 - polling_booth
 - postal_voting
 - state
 - state_level_officer
 - support_staff
 - vote
 - voters
- Trigger Functions
 - Types
 - Views
 - public
 - salesmanagement
 - Subscriptions
 - public
 - Login/Group Roles

202301475/202301475@PgSQLServer

Query History

```
-- Pending Complaints Sorted By Filing Date
SELECT * FROM Complaints
WHERE Complaint_Status='Pending'
ORDER BY Filed_Date;
```

Data Output Messages Notifications

complaint_id	complaint_type	complaint_description	complaint_status	election_id	filed_date	resolved_date	voter_id
1	CMP0000007	EVM	Pending	LS2019	2024-04-18	[null]	GHY87
2	CMP0000002	EVM	Pending	LS2019	2024-04-20	[null]	JKT23
3	CMP0000005	Staff	Pending	LS2024	2024-04-20	[null]	QWE12
4	CMP0000014	Booth	Pending	LS2024	2024-04-20	[null]	[null]
5	CMP0000010	Staff	Pending	LS2024	2024-04-21	[null]	[null]
6	CMP0000015	EVM	Pending	LS2024	2024-04-21	[null]	[null]

Showing rows: 1 to 6 Page No: 1 of 1

Successfully run. Total query runtime: 153 msec. 6 rows affected.

Total rows: 6 Query complete 00:00:00.153

--Q19. Candidate count based on Educational Qualifications

```
SELECT Educational_Qualification, count(Candidate_ID)
FROM Candidate
GROUP BY Educational_Qualification;
```

pgAdmin 4

Object Explorer

- Materialized Views
- Operators
- Sequences
- Tables (22)
 - booth_election
 - booth_level_officer
 - candidate
 - complaints
 - constituency
 - constituency_level_off
 - deployed_at
 - election
 - election_phase
 - election_schedule
 - employed_at
 - employee
 - evm_vvpat
 - national_level_officer
 - parties
 - polling_booth
 - postal_voting
 - state
 - state_level_officer
 - support_staff
 - vote
 - voters
- Trigger Functions
 - Types
 - Views
 - public
 - salesmanagement
 - Subscriptions
 - public
 - Login/Group Roles

202301475/202301475@PgSQLServer

Query History

```
-- Candidate count based on Educational Qualifications
SELECT Educational_Qualification, count(Candidate_ID)
FROM Candidate
GROUP BY Educational_Qualification;
```

Data Output Messages Notifications

educational_qualification	count
High School	36
Graduate	44
Postgraduate	35

Showing rows: 1 to 3 Page No: 1 of 1

Total rows: 3 Query complete 00:00:00.186

--Q20. National Level Officers of ECI

```
SELECT * FROM EMPLOYEE WHERE Employee_Type = 'National';
```

pgAdmin 4

File Object Tools Edit View Window Help

Object Explorer

- Materialized Views
- Operators
- Sequences
- Tables (22)
 - booth_election
 - booth_level_officer
 - candidate
 - complaints
 - constituency
 - constituency_level_off
 - deployed_at
 - election
 - election_phase
 - election_schedule
 - employed_at
 - employee
 - evm_vvpat
 - national_level_officer
 - parties
 - polling_booth
 - postal_voting
 - state
 - state_level_officer
 - support_staff
 - vote
 - voters
- Trigger Functions
- Types
- Views
- public
 - salesmanagement
- Subscriptions
- public
- Login/Group Roles

202301475/202301475@PgSQLServer

Query

```
-- National Level Officers of ECI
SELECT * FROM EMPLOYEE WHERE Employee_Type = 'National';
```

Data Output Messages Notifications

Showing rows: 1 to 3 Page No: 1 of 1

	officer_id [PK] character (10)	voter_id character (10)	designation character varying (10)	office_add text	status character varying (10)	office_contact_no character varying (10)	office_email character varying (50)	appointment_date date	tenure_end date
1	EMP0000111	BYU3456789	CEC	Booth 52, Jodhpur	Working	9876500071	neha.s@eci.gov.in	2021-12-01	2026-11-30
2	EMP0000112	VTP2345671	EC	Election HQ, Guwahati	Working	9876500072	mohit.j@eci.gov.in	2022-08-15	2027-08-14
3	EMP0000113	QWE5678912	EC	District Office, Udaipur	Working	9876500073	pooja.m@eci.gov.in	2010-03-01	2020-02-28

Total rows: 3 Query complete 00:00:00.661 CRLF Ln 2, Col 57