

# Queries

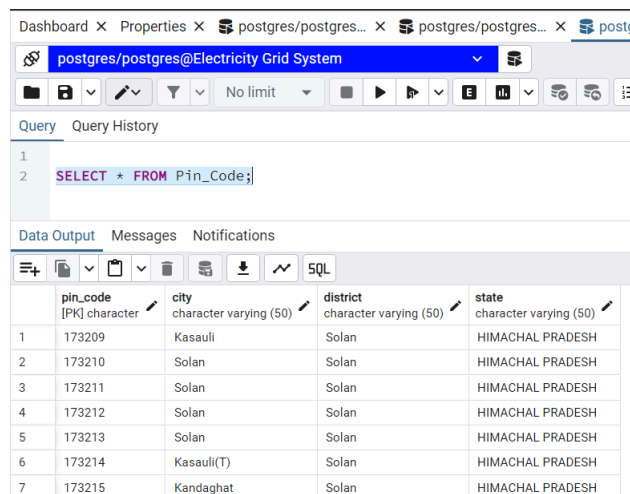
202303005- Hemil Patel

202303015- Akshat Shah

202303001- Preet Siddhapura

1) To retrieve all the pin codes ():

```
SELECT * FROM Pin_Code;
```



The screenshot shows a PostgreSQL query editor interface. The query entered is `SELECT * FROM Pin_Code;`. The results are displayed in a table with the following columns: `pin_code` (PK) character, `city` character varying (50), `district` character varying (50), and `state` character varying (50). The table contains 7 rows of data.

	pin_code [PK] character	city character varying (50)	district character varying (50)	state character varying (50)
1	173209	Kasauli	Solan	HIMACHAL PRADESH
2	173210	Solan	Solan	HIMACHAL PRADESH
3	173211	Solan	Solan	HIMACHAL PRADESH
4	173212	Solan	Solan	HIMACHAL PRADESH
5	173213	Solan	Solan	HIMACHAL PRADESH
6	173214	Kasauli(T)	Solan	HIMACHAL PRADESH
7	173215	Kandaghat	Solan	HIMACHAL PRADESH

$\pi^*$  (Pin\_Code)

2) Count the number of customers for all Pin\_Code (Customer Management):

```
SELECT
    customer.pin_code, COUNT(Customer.Customer_ID) AS Customer_Count
FROM
    Customer
GROUP BY
    customer.pin_code;
```

Query Query History	
1	SELECT customer.pin_code, COUNT(Customer.Customer_ID) AS Customer_Count
2	FROM Customer
3	GROUP BY customer.pin_code;
4	

Data Output Messages Notifications		
	pin_code character	customer_count bigint
1	281003	10
2	361012	10
3	174305	10
4	283202	10
5	283130	10
6	362226	10
7	364280	10
8	363035	10
9	284304	10
10	281004	10
11	363421	10
12	284122	10
13	283111	10

Pin\_Code  $\mathcal{H}$  COUNT(Customer\_ID) (Customer)

3)To Find all the customers in a given pin-Code which are Connected or Disconnected:  
(Customer Management)

```

SELECT
    Customer.Customer_ID, Customer.Phone_Number, Customer.Block_Flat_No,
    Customer.Street, Meter.Meter_ID
FROM
    Customer
JOIN
    Pin_Code
ON
    Customer.Pin_Code = Pin_Code.Pin_Code
JOIN
    Meter
ON
    Customer.Customer_ID=Meter.Customer_ID
WHERE
    Pin_Code.Pin_Code = '332027' AND Customer.Connection_Status='Connected';

```

Query

Query History

1

SELECT

Customer.Customer\_ID, Customer.Phone\_Number, Customer.Block\_Flat\_No, Customer.Street, Meter.Meter\_ID

2

FROM

Customer

3

JOIN

Pin\_Code ON Customer.Pin\_Code = Pin\_Code.Pin\_Code

4

JOIN

Meter ON Customer.Customer\_ID=Meter.Customer\_ID

5

WHERE

Pin\_Code.Pin\_Code = '332027'

6

AND

Customer.Connection\_Status='Connected';

7

Data Output

Messages

Notifications

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

+

$\Pi$  Customer.Customer\_ID, Customer.Phone\_Number, Customer.Block\_Flat\_No, Customer.Street, Meter.Meter\_ID ( $\sigma$  Pin\_Code.Pin\_Code = 'XYZ' AND Customer.Connection\_Status = 'Connected') ((Customer  $\bowtie$  Pin\_Code)  $\bowtie$  Meter)

4)To find all the Substations located in a given Pin-Code (Substation Management):

```

SELECT
    Substation.Substation_ID, Substation.area, Substation.status
FROM
    Substation
JOIN
    Pin_Code
ON
    Substation.Pin_Code = Pin_Code.Pin_Code
WHERE
    Pin_Code.Pin_Code = 'XYZ';
  
```

Query		Query History
1	SELECT	Substation.Substation_ID, Substation.Capacity
2	FROM	Substation
3	JOIN	Pin_Code ON Substation.Pin_Code = Pin_Code.Pin_Code
4	WHERE	Pin_Code.Pin_Code = '174405';
5		

Data Output		Messages	Notifications
	substation_id [PK] character	capacity numeric (10,2)	
1	1744050001	66.00	

$\Pi$  Substation.Substation\_ID, Substation.Capacity ( $\sigma$  Pin\_Code.Pin\_Code = 'XYZ' (Substation  $\bowtie$  Substation.Pin\_Code = Pin\_Code.Pin\_Code Pin\_Code))

5) To find the Sum of electricity load capacity in a given Pin\_Code: (Maintenance Analysis)

```

SELECT
    Pin_Code.Pin_Code, SUM(Feeder.Capacity) AS Total_Capacity
FROM
    Feeder
JOIN
    Substation
ON
    Feeder.Substation_ID = Substation.Substation_ID
JOIN
    Pin_Code
ON
    Substation.Pin_Code = Pin_Code.Pin_Code
GROUP BY
    Pin_Code.Pin_Code;
  
```

Query		Query History
1	SELECT	Pin_Code.Pin_Code, SUM(Feeder.Capacity) AS Total_Capacity
2	FROM	Feeder
3	JOIN	Substation ON Feeder.Substation_ID= Substation.Substation_ID
4	JOIN	Pin_Code ON Substation.Pin_Code = Pin_Code.Pin_Code
5	GROUP BY	Pin_Code.Pin_Code;
6		

Data Output		Messages	Notifications
pin_code	total_capacity		
[PK] character	numeric		
1	283202		58.00
2	284122		37.00
3	333029		48.00
4	332719		40.00
5	283141		51.00
6	333030		61.00
7	333031		48.00
8	283151		43.00

Pin\_Code.Pin\_Code SUM(Feeder.Capacity) (Feeder Substation.Substation\_ID (Substation Substation.Pin\_Code = Pin\_Code.Pin\_Code Pin\_Code))

## 6) List of Power sources in a given Pin\_Code: (Power Source Management)

```

SELECT
    Power_Source.Power_Source_ID, Power_Source.Type, Power_Source.Capacity
FROM
    Power_Source
JOIN
    Pin_Code
ON
    Power_Source.Pin_Code = Pin_Code.Pin_Code
WHERE
    Pin_Code.Pin_Code = 'XYZ';

```

Query		Query History
1	SELECT	Power_Source.Power_Source_ID, Power_Source.Type, Power_Source.Capacity
2	FROM	Power_Source
3	JOIN	Pin_Code ON Power_Source.Pin_Code = Pin_Code.Pin_Code
4	WHERE	Pin_Code.Pin_Code = '360001';
5		

Data Output		Messages	Notifications
power_source_id	type		capacity
[PK] character	character varying (50)		numeric (10,2)
1	THM001	Thermal	200.00

$\Pi$  Power\_Source.Power\_Source\_ID, Power\_Source.Type, Power\_Source.Capacity ( $\sigma$  Pin\_Code.Pin\_Code = 'XYZ' (Power\_Source  
 $\bowtie$  Power\_Source.Pin\_Code = Pin\_Code.Pin\_Code Pin\_Code))

7) List all the outages along with the affected areas in a given pin code: (Outage Analysis)

```

SELECT
    Outage.Outage_ID, Outage.Outage_Type, Outage.Status, Affected_Area.Area
FROM
    Outage
JOIN
    Affected_Area
ON
    Outage.Outage_ID = Affected_Area.Outage_ID AND Affected_Area.Pin_Code =
    '173213';
  
```

The screenshot shows a database query editor with a 'Query' tab selected. The query is as follows:

```

1 SELECT Outage.Outage_ID, Outage.Outage_Type, Outage.Status, Affected_Area.Area
2 FROM Outage
3 JOIN Affected_Area ON Outage.Outage_ID = Affected_Area.Outage_ID
4 and Affected_Area.Pin_Code = '173213';
5
  
```

Below the query editor, the 'Data Output' tab is selected, displaying the results of the query in a table format. The table has four columns: outage\_id, outage\_type, status, and area. The first row of data shows an outage with ID 202009280001, type Scheduled, status Ongoing, and area Area5.

	outage_id character	outage_type character varying (50)	status character varying (50)	area character varying (200)
1	202009280001	Scheduled	Ongoing	Area5

$\Pi$  Outage.Outage\_ID, Outage.Outage\_Type, Outage.Status, Affected\_Area.Area ( $\sigma$  Affected\_Area.Pin\_Code = '173213' (Outage  
 $\bowtie$  Outage.Outage\_ID = Affected\_Area.Outage\_ID Affected\_Area))

8) Find all the ongoing outages along with the affected areas in a given pin code: (Outage analysis)

```

SELECT
    Outage.Outage_ID, Outage.Outage_Type, Outage.Status, Affected_Area.Area
FROM
    Outage
LEFT JOIN
  
```

```

    Affected_Area
ON
    Outage.Outage_ID = Affected_Area.Outage_ID
WHERE
    Affected_Area.Pin_Code = 'XYZ' AND Outage.Status='Ongoing';

```

Query		Query History		
1	SELECT	Outage.Outage_ID, Outage.Outage_Type, Outage.Status, Affected_Area.Area		
2	FROM	Outage		
3	LEFT JOIN	Affected_Area ON Outage.Outage_ID = Affected_Area.Outage_ID		
4	WHERE	Affected_Area.Pin_Code = '174004'		
5	AND	Outage.Status='Ongoing';		
6				

Data Output		Messages	Notifications
<div> <div>+</div> <div>SQL</div> </div>			
outage_id	outage_type	status	area
character	character varying (50)	character varying (50)	character varying (200)
1	201903090001	Unexpected	Ongoing
			Area19

$\pi$  Outage.Outage\_ID, Outage.Outage\_Type, Outage.Status, Affected\_Area.Area ( $\sigma$  Affected\_Area.Pin\_Code = 'XYZ' AND Outage.Status = 'Ongoing' (Outage  $\bowtie$  Outage.Outage\_ID = Affected\_Area.Outage\_ID Affected\_Area))

9)Find the total Outages per Pin Code : (Outage Analysis)

```

SELECT
    COUNT(DISTINCT Outage.Outage_ID), Affected_Area.pin_code
FROM
    Outage
LEFT JOIN
    Affected_Area
ON
    Outage.Outage_ID=Affected_Area.Outage_ID
GROUP BY
    Affected_Area.Pin_Code

```

Query		Query History	
1	SELECT	count(DISTINCT	Outage.Outage_ID), Affected_Area.pin_code
2	FROM	Outage	
3	LEFT JOIN	Affected_Area ON	Outage.Outage_ID=Affected_Area.Outage_ID
4	GROUP BY	Affected_Area.Pin_Code	
5			

Data Output		Messages		Notifications	
	count		pin_code		
	bigint		character		
1	1		173209		
2	1		173210		
3	1		173211		
4	1		173212		
5	1		173213		
6	1		173214		
7	1		173215		

Affected\_Area.Pin\_Code COUNT(DISTINCT Outage.Outage\_ID) (Outage Affected\_Area.Outage\_ID = Affected\_Area.Outage\_ID)

10) Find all the feeders with total number of customers associated with it being more than its load capacity can handle (Maintenance Analysis)

```
SELECT Feeder.Feeder_ID
FROM Feeder
LEFT JOIN Meter ON Feeder.Feeder_ID=Meter.Feeder_ID
GROUP BY Feeder.Feeder_ID, Feeder.Load_Profile
HAVING count(Meter.Meter_ID) > (0.9*1000*Feeder.Load_Profile)/1.5 ;
```



Query Query History

```

1  SELECT Feeder.Feeder_ID
2  FROM Feeder
3  LEFT JOIN Meter ON Feeder.Feeder_ID=Meter.Feeder_ID
4  GROUP BY Feeder.Feeder_ID, Feeder.Load_Profile
5  HAVING count(Meter.Meter_ID) > (0.9*1000*Feeder.Load_Profile)/1.5 ;
6

```

Data Output Messages Notifications

feeder\_id  
[PK] character

Query Query History

```

1  SELECT Feeder.Feeder_ID
2  FROM Feeder
3  LEFT JOIN Meter ON Feeder.Feeder_ID=Meter.Feeder_ID
4  GROUP BY Feeder.Feeder_ID, Feeder.Load_Profile
5  HAVING count(Meter.Meter_ID) < (0.9*1000*Feeder.Load_Profile)/1.5 ;
6

```

Data Output Messages Notifications

	feeder_id [PK] character
1	333304000102
2	284122000104
3	284001000103
4	332709000101
5	283203000103
6	332707000101

$\pi$  Feeder.Feeder\_ID  
 $(\sigma \text{ COUNT(Meter.Meter\_ID) > (0.9 * 1000 * Feeder.Load\_Profile) / 1.5}$   
 $(\gamma \text{ Feeder.Feeder\_ID, Feeder.Load\_Profile, COUNT(Meter.Meter\_ID)}$   
 $(\text{Feeder} \bowtie \text{Feeder.Feeder\_ID} = \text{Meter.Feeder\_ID Meter}))$

11) Find the number of maintenance schedules of the substations which have at least 1 feeder with the total number of customers associated with it being more than its load capacity can handle. (Maintenance Analysis)

```

SELECT count(Maintenance_ID), substation_ID
FROM Maintenance_Schedule
WHERE Substation_ID IN (
    SELECT DISTINCT sub.Substation_ID
    FROM Substation AS sub
    LEFT JOIN Feeder ON sub.substation_ID = Feeder.substation_ID
    WHERE Feeder.Feeder_ID IN (
        SELECT DISTINCT Feeder.Feeder_ID
        FROM Feeder
        LEFT JOIN Meter ON Feeder.Feeder_ID=Meter.Feeder_ID
        GROUP BY Feeder.Feeder_ID, Feeder.Load_Profile
        HAVING count(Meter.Meter_ID)> (0.9*Feeder.Load_Profile)/15
    )
)
GROUP BY Substation_ID;

```

Query Query History

```

1 SELECT COUNT(Maintenance_ID) AS Maintenance_Count, Substation_ID
2 FROM Maintenance_Schedule
3 WHERE Substation_ID IN (
4     SELECT DISTINCT sub.Substation_ID
5     FROM Substation AS sub
6     LEFT JOIN Feeder ON sub.Substation_ID = Feeder.Substation_ID
7     WHERE Feeder.Feeder_ID IN (
8         SELECT Feeder.Feeder_ID
9         FROM Feeder
10        LEFT JOIN Meter ON Feeder.Feeder_ID = Meter.Feeder_ID
11        GROUP BY Feeder.Feeder_ID, Feeder.Load_Profile
12        HAVING COUNT(Meter.Meter_ID) > (0.9 * 1000 * Feeder.Load_Profile) / 1.5
13    )
14 )
15 GROUP BY Substation_ID;

```

Data Output Messages Notifications

maintenance_count	substation_id
bigint	character

$$\begin{aligned}
 &\pi_{\{Substation\_ID, COUNT(Maintenance\_ID)\}} \\
 &(\sigma_{\{Substation\_ID \in \\
 &\quad (\pi_{\{sub.Substation\_ID\}} \\
 &\quad (\sigma_{\{Feeder.Feeder\_ID \in \\
 &\quad \quad (\pi_{\{Feeder.Feeder\_ID\}} \\
 &\quad \quad (
 \end{aligned}$$

```

        σ_{COUNT(Meter.Meter_ID) > (0.9 * 1000 * Feeder.Load_Profile) / 1.5}
        (Feeder ⋈ Meter)
        GROUP BY Feeder.Feeder_ID, Feeder.Load_Profile
    )
)
}
(Substation ⋈_{Substation.Substation_ID = Feeder.Substation_ID} Feeder)
)
)}}
(Maintenance_Schedule)
) GROUP BY Substation_ID

```

12) List all the pincodes having had more than 10 outages in the last week (Outage Analysis)

```

SELECT
    pc.Pin_Code, COUNT(ou.Outage_ID) AS Outage_Count
FROM
    Outage as ou
NATURAL JOIN
    affected_area as aa
NATURAL JOIN
    pin_code as pc
WHERE
    ou.start_date_time BETWEEN '2017-01-01' AND '2023-06-01'
GROUP BY
    pc.Pin_Code
HAVING
    COUNT(Outage_ID) > 10;

```



Query		Query History
1	SELECT	p2.Power_Source_ID, p2.owner_id
2	FROM	Power_Source as p1
3	JOIN	Power_Source as p2 ON p1.Owner_ID = p2.Owner_ID
4	and	p1.Power_Source_ID = 'THM001';
5		

Data Output		Messages	Notifications
	power_source_id [PK] character	owner_id character	
1	THM001	SEL	
2	THM002	SEL	
3	THM003	SEL	

$\pi_{\text{Power\_Source\_ID, Owner\_ID}}(\sigma_{\text{Power\_Source\_ID}='THM001'}(\text{Power\_Source}) \bowtie_{\text{Owner\_ID}} \text{Power\_Source})$

14) Find the average electricity rate for each customer type over all time. (Consumption Trend Analysis)

```

SELECT
    type_name, AVG(electricity_Rate) AS Average_Rate
FROM
    Electricity_Rate
JOIN
    customer_type
ON
    electricity_rate.customer_type_id = customer_type.customer_type_id
GROUP BY
    type_name;

```

Query		Query History
1	SELECT	type_name, AVG(electricity_Rate) AS Average_Rate
2	FROM	Electricity_Rate join customer_type on electricity_rate.customer_type_id = customer_type.customer_type_id
3	GROUP BY	type_name;
4		

Data Output		Messages	Notifications
	type_name character varying (100)	average_rate numeric	
1	Agricultural	7.9840000000000000	
2	Industrial	18.666666666666667	
3	Transportation	7.9020000000000000	
4	Commercial	12.9940000000000000	
5	Residential	9.7800000000000000	
6	Institutional	10.784666666666667	

$\pi_{\text{type\_name, AVG(electricity\_rate)} \rightarrow \text{Average\_Rate}}(\pi_{\text{type\_name, electricity\_rate}}(\text{Electricity\_Rate} \bowtie_{\text{customer\_type\_id}} \text{customer\_type}))$

15) Find the total capacity of power sources state wise. (Power Source Management)

```
SELECT
    p.State, SUM(capacity) AS Total_Capacity
FROM
    Power_Source as s
JOIN
    Pin_code as p
ON
    s.pin_code = p.pin_code
GROUP BY
    p.State;
```

Query Query History

```
1 SELECT p.State, SUM(capacity) AS Total_Capacity
2 FROM Power_Source as s join Pin_code as p on s.pin_code = p.pin_code
3 GROUP BY p.State;
4
```

Data Output Messages Notifications

	state character varying (50)	total_capacity numeric
1	GUJARAT	1670.50

State  $\xrightarrow{\text{SUM(capacity)} \rightarrow \text{Total\_Capacity}}$   $(\pi_{\text{State, capacity}}(\sigma_{\text{pin\_code}}(\text{Power\_Source} \bowtie \text{Pin\_code})))$

16) Find the total capacity of power sources, type wise. (Power Source Management)

```
SELECT
    Type, SUM(Capacity) AS Total_Capacity
FROM
    Power_Source
GROUP BY
    Type;
```

Query		Query History
1	SELECT	Type, SUM(Capacity) AS Total_Capacity
2	FROM	Power_Source
3	GROUP BY	Type;
4		
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>SQL</div> </div>		
	type	total_capacity
	character varying (50)	numeric
1	Wind	330.50
2	Thermal	660.00
3	Hydro	445.00
4	Solar	235.00

$\pi_{Type} \sigma_{SUM(Capacity) \rightarrow Total\_Capacity}(\pi_{Type,Capacity}(Power\_Source))$

17)Find customers with unpaid bills over a certain amount (Revenue Analysis)

```

SELECT
    C.Customer_ID, C.Customer_Name, B.Total_Price, B.Billing_Date
FROM
    Customer AS C
JOIN
    Meter AS M
ON
    C.Customer_ID = M.Customer_ID
JOIN
    Bill AS B
ON
    M.Meter_ID = B.Meter_ID
WHERE
    B.Payment_Status = 'Unpaid' AND B.Total_Price > 1000;

```

Query

Query History

12

13

14

15

16

17

SELECT C.Customer\_ID, C.Customer\_Name, B.Total\_Price, B.Billing\_Date  
FROM Customer AS C JOIN Meter AS M ON C.Customer\_ID = M.Customer\_ID  
JOIN Bill AS B ON M.Meter\_ID = B.Meter\_ID WHERE  
B.Payment\_Status = 'Unpaid' AND B.Total\_Price > 1000

Data Output

Messages

Notifications

SQL

	customer_id character varying (12)	customer_name character varying (200)	total_price numeric (10,2)	billing_date date
1	332027000007	Shivam Mahajan	2804.00	2022-09-19
2	332029000003	Tanish Reddy	1125.00	2021-02-12
3	332030000005	Pihu Sharma	2578.00	2023-07-01
4	332031000005	Hemil Mahajan	3383.00	2021-01-24
5	332303000003	Anya Murthy	3031.00	2023-11-21
6	332303000006	Nisha Mukherjee	1740.00	2023-03-19
7	332303000007	Sia Jha	2331.00	2023-09-02
8	332315000002	Anya Mukherjee	2041.00	2022-10-15
9	332315000007	Harsh Bajaj	3108.00	2022-03-11
10	332316000002	Advik Malhotra	1057.00	2023-08-22
11	332316000006	Shivansh Murthy	2775.00	2023-01-02
12	332317000004	Krish Acharya	2024.00	2021-07-24
13	332318000004	Ishaan Vyas	2947.00	2021-03-01
14	332318000010	Aarush Chaudhary	2932.00	2021-05-19
15	332404000003	Isha Adiga	1463.00	2021-09-15

$\Pi_{\text{Customer\_ID, Customer\_Name, Total\_Price, Billing\_Date}}(\sigma_{\text{Payment\_Status}='Unpaid' \text{ AND } \text{Total\_Price}>1000}((\text{Customer} \bowtie_{\text{Customer\_ID}} \text{Meter}) \bowtie_{\text{Meter\_ID}} \text{Bill}))$

18) List active meters in substations with more than 4 feeders (Substation Management)

```

SELECT
    M.Meter_ID, M.Current_Reading, S.Substation_ID
FROM
    Meter AS Mx
JOIN
    Feeder AS F
ON
    M.Feeder_ID = F.Feeder_ID
JOIN
    Substation AS S
ON
    F.Substation_ID = S.Substation_ID
WHERE
    S.Substation_ID IN (
        SELECT
            Substation_ID
        FROM
            Feeder
        GROUP BY

```



```

        Substation_ID
    HAVING
        COUNT(Feeder_ID) > 4
    ) AND M.Status = 'Active';

```

Query

Query History

11

12

13

14

15

16

17

SELECT M.Meter\_ID, M.Current\_Reading, S.Substation\_ID FROM Meter AS M JOIN Feeder AS F  
ON M.Feeder\_ID = F.Feeder\_ID JOIN Substation AS S ON F.Substation\_ID = S.Substation\_ID  
WHERE S.Substation\_ID IN (SELECT Substation\_ID FROM Feeder GROUP BY  
Substation\_ID HAVING COUNT(Feeder\_ID) > 4) AND M.Status = 'Active';

Data Output

Messages

Notifications

	meter_id character	current_reading numeric (10,2)	substation_id character
1	202007250001	1862.00	3330250001
2	201907090001	2702.00	2832030001
3	201704170001	2534.00	2831510001
4	202012050001	6606.00	3330360001
5	201505140001	336.00	3327050001
6	201611240001	9219.00	3327420001
7	201801240001	8917.00	3330310001
8	201609010001	3511.00	2841270001
9	201711270001	9791.00	3335020001
10	201905220001	1835.00	3330110001
11	201801140001	6715.00	3330300001
12	201802240001	3200.00	3327130001
13	201702100001	5708.00	3330210001
14	201810200001	3332.00	3330360001

$\pi_{\text{Meter\_ID, Current\_Reading, Substation\_ID}}(\sigma_{\text{Substation\_ID} \in (\pi_{\text{Substation\_ID}}(\sigma_{\text{COUNT(Feeder\_ID)} > 4}(\text{Substation\_ID} \bowtie \text{COUNT(Feeder\_ID)(Feeder)}))})$   
 AND Status='Active' ((Meter  $\bowtie_{\text{Feeder\_ID}}$  Feeder)  $\bowtie_{\text{Substation\_ID}}$  Substation))

19) Find substations connected to a specific substation (Substation Management)

```

SELECT
    SC.Substation2_ID
FROM
    Substation_Connections AS SC
WHERE
    Substation1_ID = '1745050001';

```

Query		Query History
26	SELECT	
27	SC.Substation2_ID	
28	FROM	
29	Substation_Connections AS SC	
30	WHERE	
31	Substation1_ID = '1745050001';	
Data Output		Messages
		Notifications
	substation2_id	
	character	
1	3333040001	
2	3327050001	
3	3330310001	

$\Pi_{SC.Substation2\_ID} (\sigma_{SC.Substation1\_ID = 'SUB1234'} (Substation\_Connections))$

20) List of Total revenue generated by each customer type in a given year. (Revenue Analysis)

```

SELECT
    SUM(Total_Price), bier.customer_type_id
FROM
    (Bill as bi JOIN electricity_rate as er ON er.rate_id = bi.rate_id) as bier
JOIN
    customer_type as ct ON ct.customer_type_id = bier.customer_type_id
WHERE
    bier.payment_status = 'Paid' and DATE_PART('year', bier.billing_date) = 2021
GROUP BY
    bier.customer_type_id;

```

Query	Query History
23	
24	SELECT
25	SUM(Total_Price), bier.customer_type_id
26	FROM
27	(Bill as bi JOIN electricity_rate as er ON er.rate_id = bi.rate_id) as bier
28	JOIN
29	customer_type as ct ON ct.customer_type_id = bier.customer_type_id
30	WHERE
31	bier.payment_status = 'Paid' and DATE_PART('year', bier.billing_date) = 2021
32	GROUP BY
33	bier.customer_type_id;
34	
Data Output	Messages Notifications
	SQL
	sum numeric customer_type_id integer
1	3156617.00 1
2	713801.00 2
3	673355.00 3
4	593536.00 4
5	621399.00 5
6	616930.00 6

customer\_type\_id  $\rightarrow$  SUM(Total\_Price)  $\rightarrow$  Total\_Price ( $\sigma_{\text{payment\_status}='Paid' \text{ AND } \text{YEAR}(\text{billing\_date})=2021}$ ) ((Bill  $\bowtie_{\text{rate\_id}}$  electricity\_rate)  $\bowtie_{\text{customer\_type\_id}}$  customer\_type))

21) Find the Customer Distribution by customer type in every region (Customer Management)

```

SELECT
    ct.Type_Name AS Customer_Type,
    p.City,
    p.District,
    COUNT(c.Customer_ID) AS Customer_Count
FROM
    customer as c
JOIN
    Customer_Type as ct ON c.Customer_Type_ID = ct.Customer_Type_ID
JOIN
    Pin_Code as p ON c.Pin_Code = p.Pin_Code
GROUP BY
    ct.Customer_Type_ID, p.City, p.District
ORDER BY
    Customer_Count DESC;

```

Query

Query History

21

22

23

24

25

26

27

SELECT ct.Type\_Name AS Customer\_Type,p.City,p.District,  
COUNT(c.Customer\_ID) AS Customer\_Count FROM customer as c  
JOIN Customer\_Type as ct ON c.Customer\_Type\_ID = ct.Customer\_Type\_ID  
JOIN Pin\_Code as p ON c.Pin\_Code = p.Pin\_Code  
GROUP BY ct.Customer\_Type\_ID, p.City, p.District  
ORDER BY Customer\_Count DESC;

Data Output

Messages

Notifications

	customer_type character varying (100)	city character varying (50)	district character varying (50)	customer_count bigint
1	Residential	Rajkot	Rajkot	110
2	Residential	Mathura	Mathura	70
3	Residential	Jamnagar	Jamnagar	70
4	Residential	Jhansi	Jhansi	60
5	Residential	Fatehpur	Sikar	55
6	Residential	Sikar	Sikar	55
7	Residential	Bhavnagar	Bhavnagar	55
8	Residential	Chirawa	Jhujhunu	50
9	Residential	Agra	Agra	50
10	Residential	Nimkathana	Sikar	50
11	Residential	Kalavad	Jamnagar	40
12	Residential	Bikaner	Bikaner	40
13	Residential	Ghumarwin	Bilaspur (HP)	40
14	Residential	Khetri	Jhujhunu	40
15	Residential	Palanpur	Rajkot	25

Total rows: 960 of 960 Query complete 00:00:00 263

$\pi_{Customer\_Count \text{ DESC}}(\pi_{Type\_Name, City, District} \sigma_{COUNT(Customer\_ID) \rightarrow Customer\_Count}(\pi_{Type\_Name, City, District, Customer\_ID} ((customer \bowtie_{Customer\_Type\_ID} Customer\_Type) \bowtie_{Pin\_Code} Pin\_Code)))$

22) Get the number of customers per state who had a meter installed before Certain Date.  
(Customer Management)

```

SELECT
    State, count(cu.customer_id)
FROM
    meter AS me
JOIN
    customer AS cu
ON
    me.customer_id = cu.customer_id
JOIN
    pin_code as pc
ON
    cu.pin_code = pc.pin_code
WHERE
    me.installation_date < '2019-01-01'
GROUP BY
    pc.state;

```

Query

Query History

23

24

25

26

27

28









29

SELECT State, count(cu.customer\_id)FROM meter AS me JOIN  
customer AS cu ON me.customer\_id = cu.customer\_id  
JOIN pin\_code as pc ON cu.pin\_code = pc.pin\_code  
WHERE me.installation\_date < '2019-01-01'  
GROUP BY pc.state;

Data Output

Messages

Notifications



SQL

state

character varying (50)

count

bigint

1

GUJARAT

1316

2

UTTAR PRADESH

693

3

HIMACHAL PRADESH

694

4

RAJASTHAN

689

5

DAMAN & DIU

18

State  $\xrightarrow{\pi}$  COUNT(customer\_id)  $\rightarrow$  Customer\_Count  $(\pi_{State, customer\_id} (\sigma_{installation\_date < '2019-01-01'} ((meter \bowtie_{customer\_id} customer) \bowtie_{pin\_code} pin\_code)))$

- 23) Retrieve the count of electricity meters in a particular state, categorized by the type of customer. (Meter Management)

```

SELECT
    COUNT(me.meter_id) as number_of_meters, ct.type_name
FROM
    meter as me
NATURAL JOIN
    Customer as cu
NATURAL JOIN
    Customer_Type as ct
NATURAL JOIN
    Pin_code as pc
WHERE
    pc.state = 'GUJARAT'
GROUP BY
    ct.type_name
ORDER BY
    number_of_meters DESC;

```

Query	Query History
22	
23	SELECT COUNT(me.meter_id) as number_of_meters, ct.type_name
24	FROM meter as me NATURAL JOIN Customer as cu
25	NATURAL JOIN Customer_Type as ct NATURAL JOIN
26	Pin_code as pc WHERE pc.state = 'GUJARAT'
27	GROUP BY ct.type_name ORDER BY number_of_meters DESC;
28	
Data Output	Messages Notifications
+	SQL
number_of_meters bigint	type_name character varying (100)
1	1094 Residential
2	202 Transportation
3	199 Commercial
4	198 Agricultural
5	198 Institutional
6	197 Industrial

$\pi_{\text{number\_of\_meters DESC}}(\sigma_{\text{state='GUJARAT'}}(\text{meter} \bowtie \text{Customer} \bowtie \text{Customer\_Type} \bowtie \text{Pin\_code}))$

24) Get the count of connected customers in each city within a particular state. (Customer Management)

```

SELECT
    COUNT(customer_id) AS NUMBER_OF_CUSTOMERS, CITY, STATE
FROM
    customer
NATURAL JOIN
    pin_code
WHERE
    connection_status='Connected' AND STATE='GUJARAT'
GROUP BY
    city, STATE
ORDER BY
    NUMBER_OF_CUSTOMERS DESC;

```

Query Query History

```

22 v SELECT COUNT(customer_id) AS NUMBER_OF_CUSTOMERS, CITY, STATE
23 FROM customer NATURAL JOIN pin_code WHERE
24 connection_status='Connected' AND STATE='GUJARAT' GROUP BY
25 city, STATE ORDER BY NUMBER_OF_CUSTOMERS DESC;

```

Data Output Messages Notifications

	number_of_customers bigint	city character varying (50)	state character varying (50)
1	140	Jamnagar	GUJARAT
2	120	Rajkot	GUJARAT
3	110	Bhavnagar	GUJARAT
4	80	Kalavad	GUJARAT
5	70	Jamjodhpur	GUJARAT
6	70	Gondal	GUJARAT
7	60	Dhoraji	GUJARAT
8	60	Junagadh	GUJARAT
9	60	Porbandar	GUJARAT
10	50	Keshod	GUJARAT
11	50	Limbdi	GUJARAT
12	50	Mangrol	GUJARAT
13	50	Patan-Veraval	GUJARAT
14	50	Wadhwan city	GUJARAT
15	50	Jasdan	GUJARAT
16	50	Una	GUJARAT

Total rows: 65 of 65 Query complete 00:00:00.110

$\pi_{\text{NUMBER\_OF\_CUSTOMERS DESC}(\text{city, state})}(\sigma_{\text{connection\_status='Connected' AND state='GUJARAT'}}(\text{customer} \bowtie \text{pin\_code}))$

25) Get the average amount of bill for each customer type for a particular State (Revenue Analysis)

```

SELECT
    AVERAGE_TOTAL_PRICE, type_name
FROM
    (
        SELECT
            AVG(total_price) AS AVERAGE_TOTAL_PRICE, customer_type_id
        FROM
            bill
        NATURAL JOIN
            meter
        NATURAL JOIN
            (SELECT
                customer_id, state, customer_type_id

```

```

FROM
    customer AS cs
JOIN
    pin_code AS pc
ON
    cs.pin_code= pc.pin_code AND pc.state='GUJARAT'
)
GROUP BY customer_type_id
)
NATURAL JOIN
    customer_type
ORDER BY
    AVERAGE_TOTAL_PRICE DESC;

```

Query Query History

```

31 SELECT AVERAGE_TOTAL_PRICE, type_name FROM (
32 SELECT AVG(total_price) AS AVERAGE_TOTAL_PRICE, customer_type_id
33 FROM bill NATURAL JOIN meter NATURAL JOIN (SELECT
34 customer_id, state, customer_type_id FROM customer AS cs
35 JOIN pin_code AS pc ON cs.pin_code= pc.pin_code AND pc.state='GUJARAT')
36 GROUP BY customer_type_id) NATURAL JOIN customer_type ORDER BY
37 AVERAGE_TOTAL_PRICE DESC;

```

Data Output Messages Notifications

	average_total_price numeric	type_name character varying (100)
1	2019.9924242424242424	Institutional
2	2009.0841584158415842	Transportation
3	1985.3197969543147208	Industrial
4	1970.4808853118712274	Residential
5	1957.7738693467336683	Commercial
6	1916.8611111111111111	Agricultural

$\pi_{\text{customer\_id, state, customer\_type\_id}} (\sigma_{\text{state}='GUJARAT'} (\text{customer} \bowtie \text{pin\_code}))) \bowtie \text{customer\_type}$   
 $\rightarrow \text{AVERAGE\_TOTAL\_PRICE} (\text{bill} \bowtie \text{meter} \bowtie$

26)

A] Find Average Electrical energy consumption by all the meters connected to each substation yearly. (Consumption Trend Analysis)

```

SELECT
    avg(Bill.Total_Price/er.Electricity_Rate) , sub.Substation_ID,
    EXTRACT (YEAR FROM bill.Billing_Date)
FROM
    Feeder AS fe
JOIN

```



```

        Substation AS sub
ON
        sub.substation_ID = fe.substation_ID
JOIN
        Meter
ON
        Meter.Feeder_ID = fe .Feeder_ID
JOIN
        Bill
ON
        Bill.Meter_ID= Meter.Meter_ID
JOIN
        Electricity_Rate AS er
ON
        Bill.rate_ID = er.rate_ID
GROUP BY
        sub.Substation_ID, EXTRACT (YEAR FROM bill.Billing_Date)

```

Query

Query History

31

▼

```
SELECT avg(Bill.Total_Price/er.Electricity_Rate) , sub.Substation_ID, EXTRACT (YEAR FROM bill.Billing_Date)
FROM Feeder AS fe
JOIN Substation AS sub ON sub.substation_ID = fe.substation_ID
JOIN Meter ON Meter.Feeder_ID = fe .Feeder_ID
JOIN Bill ON Bill.Meter_ID= Meter.Meter_ID
JOIN Electricity_Rate AS er ON Bill.rate_ID = er.rate_ID
GROUP BY sub.Substation_ID, EXTRACT (YEAR FROM bill.Billing_Date);
```

29

Data Output

Messages

Notifications

≡

📄

▼

🗑

🔍

📥

📈

SQL

	avg numeric	substation_id [PK] character	extract numeric
1	128.4106473517297727	3327010001	2023
2	107.7466213060428725	3330530001	2023
3	131.9903657211321428	2843010001	2021
4	108.9484130155559467	3327050001	2022
5	147.7427540389720067	2831420001	2023
6	120.7870701720464690	3326010001	2023
7	134.4653453581281742	3327020001	2022
8	139.4053556877897279	2841250001	2023
9	125.7420860622860592	3327050001	2023
10	111.3282590953449613	3333020001	2023
11	107.1800607105658791	3335140001	2022
12	135.4976365663874526	2831520001	2022
13	114.9507802936239474	2841280001	2021
14	120.4274564455780564	3330220001	2023
Total rows: 300 of 300		Query complete 00:00:00.155	

$$\text{Substation\_ID, Year} \xrightarrow{\pi} \text{AVG}(\text{Total\_Price} / \text{Electricity\_Rate}) \rightarrow \text{Average\_Consumption} (\pi_{\text{Total\_Price, Electricity\_Rate, Substation.Substation\_ID, EXTRACT(YEAR FROM Billing\_Date)}} \rightarrow \text{Year} (((\text{Feeder} \bowtie_{\text{Substation\_ID}} \text{Substation}) \bowtie_{\text{Meter}} \text{Meter}) \bowtie_{\text{Meter\_ID}} \text{Bill}) \bowtie_{\text{Rate\_ID}} \text{Electricity\_Rate}))$$

B] Find Average Electrical energy consumption by all the meters connected to a given substation for a given year. (Consumption Trend Analysis)

```
SELECT
    avg(Bill.Total_Price/er.Electricity_Rate)
FROM
    Feeder AS fe
JOIN
    Substation AS sub
ON
    sub.substation_ID = fe.substation_ID
JOIN
    Meter
ON
    Meter.Feeder_ID = fe.Feeder_ID
JOIN
    Bill
ON
    Bill.Meter_ID= Meter.Meter_ID
JOIN
    Electricity_Rate AS er
ON
    Bill.rate_ID = er.rate_ID
WHERE
    sub.Substation_ID='xyz' AND DATE_PART('year', bier.billing_date) = yr
```

27)List the total power generation from each power source owner with active power sources (Power Source Management)

```
SELECT
    SUM(ps.generation_data) AS Total_generation ,pso.name
FROM
    power_source AS ps
JOIN
    power_source_owner AS PSO
ON
    ps.status = 'Active' AND ps.owner_id = PSO.owner_id
GROUP BY
    PSO.name
ORDER BY
    Total_generation DESC;
```

Query	Query History
27	
28	SELECT SUM(ps.generation_data) AS Total_generation ,pso.name
29	FROM power_source AS ps JOIN power_source_owner AS pso
30	ON ps.status = 'Active' AND ps.owner_id = pso.owner_id
31	GROUP BY pso.name ORDER BY Total_generation DESC;
32	

Data Output	Messages	Notifications
<div> <div>+</div> <div>📄</div> <div>▼</div> <div>📋</div> <div>▼</div> <div>🗑️</div> <div>🔍</div> <div>⬇️</div> <div>📈</div> <div>SQL</div> </div>		
	total_generation numeric	name character varying (200)
1	9200.00	Suzlon Energy
2	6700.00	NTPC Limited
3	4700.00	Tata Power
4	3100.00	Adani Green Energy

$\pi_{Total\_generation\ DESC}(name \bowtie SUM(generation\_data) \rightarrow Total\_generation (\pi_{generation\_data, name}(\sigma_{status='Active'}(power\_source) \bowtie power\_source\_owner)))$

28) Count of maintenance teams categorized by their type. (Maintenance Team Analysis)

SELECT  
     COUNT(team\_id),team\_type  
 FROM  
     maintenance\_team  
 GROUP BY  
     Team\_type;

Query	Query History
27	SELECT
28	COUNT(team_id),team_type
29	FROM
30	maintenance_team
31	GROUP BY
32	Team_type;

Data Output	Messages	Notifications
<div> <div>+</div> <div>📄</div> <div>▼</div> <div>📋</div> <div>▼</div> <div>🗑️</div> <div>🔍</div> <div>⬇️</div> <div>📈</div> <div>SQL</div> </div>		
	count bigint	team_type character varying (100)
1	25	Emergency Response Team
2	35	Health and Safety Team
3	40	Metering Team
4	70	Substation Maintenance Team
5	40	Transmission Line Maintenance Team

$team\_type \bowtie COUNT(team\_id) \rightarrow team\_count (\pi_{team\_id, team\_type}(maintenance\_team))$

29)The average salary spent on maintenance teams, categorized by their types. (Maintenance Team Analysis)

```

SELECT
    AVG(team_salary),team_type
FROM
    (
        SELECT
            SUM(salary) AS team_salary, team_type
        FROM
            employee
        NATURAL JOIN
            Maintenance_team
        GROUP BY
            team_id,team_type
    )
GROUP BY team_type;

```

Query

Query History

32

33 ▼ SELECT AVG(team\_salary),team\_type FROM (

34 SELECT SUM(salary) AS team\_salary, team\_type

35 FROM employee NATURAL JOIN Maintenance\_team

36 GROUP BY team\_id,team\_type)GROUP BY team\_type;

37

38

Data Output

Messages

Notifications

team\_type  $\xrightarrow{f}$  AVG(team\_salary)  $\rightarrow$  avg\_team\_salary ( $\Pi_{team\_salary, team\_type} (team\_id, team\_type \xrightarrow{f} SUM(salary) \rightarrow team\_salary (\Pi_{salary, team\_type, team\_id} (employee \bowtie Maintenance\_team)))$ )

30) The number of substations based on type of circuit breakers they contain (Substation Management)

```

SELECT
    COUNT(substation_id) AS number_of_substations, circuit_breakers
FROM
    substation
GROUP BY
    circuit_breakers;

```

Query	Query History
40	SELECT
41	COUNT(substation_id) AS number_of_substations,circuit_breakers
42	FROM
43	substation
44	GROUP BY
45	circuit_breakers;
46	
Data Output	Messages
Notifications	
	SQL
	number_of_substations bigint
	circuit_breakers character varying (50)
1	27 ACB
2	18 HVBC
3	23 MCCB
4	13 MCB
5	12 OBC
6	23 SF6
7	10 VCB
8	25 HCB

circuit\_breakers  $\pi_{\text{circuit\_breakers}}$  COUNT(substation\_id)  $\rightarrow$  number\_of\_substations ( $\pi_{\text{substation\_id,circuit\_breakers}}$ (substation))

31) Get the Substation id's substations where there are more than one "Manager". (Substation Management)

```

SELECT
    substation_id
FROM
    (
        SELECT
            COUNT(EMPLOYEE_ID) AS ne,substation_id
        FROM
            employee
        WHERE
            role='Manager'
        GROUP BY
            Substation_id
    )
WHERE
    ne>1 ;

```

Query		Query History
39		
40	SELECT	substation_id FROM (SELECT
41	COUNT(EMPLOYEE_ID) AS	ne,substation_id
42	FROM employee WHERE role='Manager' GROUP BY	Substation_id) WHERE ne>1 ;
43		
44		
Data Output		Messages Notifications
	substation_id	
	character	
1	3326010001	
2	3327140001	
3	2841260001	
4	3330330001	
5	3326020001	
6	3330340001	
7	3326030001	
8	3337010001	

$\pi_{\text{substation\_id}}(\sigma_{\text{ne}>1}(\text{substation\_id} \bowtie \text{COUNT}(\text{employee\_id}) \rightarrow \text{ne}) (\pi_{\text{employee\_id}, \text{substation\_id}}(\sigma_{\text{role}='Manager'}(\text{employee}))))))$

32) To find the electrical consumption per Customer Type year-wise. (Consumption Trend Analysis)

```
SELECT avg(Bill.Total_Price/er.Electricity_Rate) , er.customer_type_ID AS ctp, EXTRACT
(YEAR FROM bill.Billing_Date) AS yr
FROM Bill
JOIN Electricity_Rate AS er ON Bill.rate_ID = er.rate_ID
GROUP BY er.customer_type_ID, EXTRACT (YEAR FROM bill.Billing_Date)
ORDER BY ctp,yr;
```



Query

Query History

50

SELECT

avg(Bill.Total\_Price/er.Electricity\_Rate) , er.customer\_type\_ID AS ctp, er.electricity\_rate AS rate,

51

EXTRACT (YEAR FROM bill.Billing\_Date) AS yr

52

FROM Bill

53

JOIN Electricity\_Rate AS er ON Bill.rate\_ID = er.rate\_ID

54

GROUP BY er.customer\_type\_ID, er.electricity\_rate, EXTRACT (YEAR FROM bill.Billing\_Date)

55

ORDER BY ctp,rate,yr;

Data Output

Messages

Notifications

SQL

	avg numeric	ctp integer	rate numeric (10,2)	yr numeric
1	125.7339811878071811	1	16.04	2021
2	123.0399682880580281	1	16.04	2022
3	123.9711402518151652	1	16.04	2023
4	117.9164634339924325	1	16.84	2021
5	116.1384658376414699	1	16.84	2022
6	115.7884844046302974	1	16.84	2023
7	110.7544836116264688	2	18.48	2021
8	100.7050203992650036	2	18.48	2022
9	104.9251059224463480	2	18.48	2023
10	111.9111149365926436	2	18.84	2021
11	103.2122478768577495	2	18.84	2022
12	104.9853273290329560	2	18.84	2023
13	67.6882481951932742	3	28.24	2021
14	68.7492578582212346	3	28.24	2022
15	68.9485363550519358	3	28.24	2023

Total rows: 36 of 36

Query complete 00:00:00.140

$T_{ctp, rate, yr(ctp, rate, yr)} \xrightarrow{AVG(price\_rate)} avg\_price\_rate$   
 $(\Pi_{Bill.Total\_Price/er.Electricity\_Rate \rightarrow price\_rate, er.customer\_type\_ID \rightarrow ctp, er.electricity\_rate \rightarrow rate, YEAR(Bill.Billing\_Date) \rightarrow yr} (\sigma_{Bill.rate\_ID=er.rate\_ID} (Bill \times Electricity\_Rate))))$

34) To find the Areas which are being affected by outages more frequently than the average outages per area for a given year. Subsequently providing the Outage Count and Average Time duration for that area too. (Outage Analysis)

```

WITH OutagesPerArea AS(
SELECT aa.area As areaName, Count(Distinct aa.outage_ID) As OutageCount,
AVG(EXTRACT(EPOCH FROM (o.End_Date_Time - o.Start_Date_Time))) / 3600) AS
AvgOutageDuration
From Affected_Area As aa
Join Outage AS o ON o.Outage_ID = aa.Outage_ID
Where EXTRACT (YEAR FROM o.Start_Date_Time) = 2020
Group By aa.area
),
AverageOutagePerArea AS(
SELECT Avg(OutageCount) As AvgOutage
From OutagesPerArea
)

```



```

SELECT opa.areaName, opa.OutageCount, opa.AvgOutageDuration
From OutagesPerArea As opa
Join AverageOutagePerArea AS Averg ON opa.OutageCount >= Averg.AvgOutage
ORDER by opa.areaName

```

	areaname character varying (200)	outagecount bigint	avgoutageduration numeric
1	Area1	1	3.2166666666666667
2	Area12	1	1.8500000000000000
3	Area2	1	3.2500000000000000
4	Area20	1	2.1166666666666667
5	Area26	1	0.8500000000000000
6	Area28	1	5.7500000000000000
7	Area32	1	0.3166666666666667
8	Area41	1	2.4000000000000000
9	Area46	1	3.8833333333333333
10	Area5	1	7.0166666666666667

35) To find the number of Customers located in a pin code over the years in those pin codes whose average electrical consumption per customer in any given year is less than 'x' units.

```

WITH YearlyCustomerConsumption AS (
SELECT
    c.Pin_Code,
    EXTRACT(YEAR FROM b.Billing_Date) AS Year,
    c.Customer_ID,
    SUM(b.Total_Price/er.Electricity_rate) AS YearlyCustomerUsage
FROM
    Bill b
JOIN
    Meter As m ON b.Meter_ID = m.Meter_ID
JOIN
    Customer c ON c.Customer_ID = m.Customer_ID
JOIN
    Electricity_rate er ON er.rate_ID = b.rate_ID

```

```
GROUP BY
    c.Pin_Code, EXTRACT(YEAR FROM b.Billing_Date), c.customer_id
),
```

```
AvgConsumptionPerPinCode AS (
    SELECT
        Pin_Code,
        Year,
        AVG(YearlyCustomerUsage) AS AvgConsumptionPerCustomer
    FROM
        YearlyCustomerConsumption
    GROUP BY
        Pin_Code, Year
    HAVING
        AVG(YearlyCustomerUsage) < 100
),
```

```
MeterCountPerYear AS (
    SELECT
        c.Pin_Code,
        EXTRACT(YEAR FROM b.Billing_Date) AS Year,
        COUNT(DISTINCT c.customer_ID) AS CustomerCount
    FROM
        Bill b
    JOIN
        Meter As m ON b.Meter_ID = m.Meter_ID
    JOIN
        Customer c ON m.Customer_ID = c.Customer_ID
    WHERE
        c.Pin_Code IN (SELECT Pin_Code FROM AvgConsumptionPerPinCode)
    GROUP BY
        c.Pin_Code, EXTRACT(YEAR FROM b.Billing_Date)
)
```

```
SELECT
    Pin_Code,
    Year,
    CustomerCount
FROM
    MeterCountPerYear
ORDER BY
    Pin_Code, Year;
```

Query Query History

```

31
32 WITH YearlyCustomerConsumption AS (SELECT c.Pin_Code,EXTRACT(YEAR FROM b.Billing_Date) AS Year,c.Customer_ID,
33 SUM(b.Total_Price/er.Electricity_rate) AS YearlyCustomerUsage FROM Bill b JOIN Meter As m ON b.Meter_ID = m.Meter_ID
34 JOIN Customer c ON c.Customer_ID = m.Customer_ID JOIN Electricity_rate er ON er.rate_ID = b.rate_ID GROUP BY
35 c.Pin_Code, EXTRACT(YEAR FROM b.Billing_Date), c.customer_id),AvgConsumptionPerPinCode AS (SELECT Pin_Code,Year,
36 AVG(YearlyCustomerUsage) AS AvgConsumptionPerCustomer FROM YearlyCustomerConsumption GROUP BY Pin_Code, Year HAVING
37 AVG(YearlyCustomerUsage) < 100),MeterCountPerYear AS (SELECT c.Pin_Code,EXTRACT(YEAR FROM b.Billing_Date) AS Year,
38 COUNT(DISTINCT c.customer_ID) AS CustomerCount FROM Bill b JOIN Meter As m ON b.Meter_ID = m.Meter_ID JOIN
39 Customer c ON m.Customer_ID = c.Customer_ID WHERE c.Pin_Code IN (SELECT Pin_Code FROM AvgConsumptionPerPinCode)
40 GROUP BY c.Pin_Code, EXTRACT(YEAR FROM b.Billing_Date))SELECT Pin_Code,Year,CustomerCount
41 FROM MeterCountPerYear ORDER BY Pin_Code, Year;
42

```

Data Output Messages Notifications

	pin_code	year	customercount
	character	numeric	bigint
1	173221	2021	5
2	173221	2022	8
3	173221	2023	1
4	174003	2021	5
5	174003	2022	8
6	174003	2023	4
7	174005	2021	6
8	174005	2022	7
9	174005	2023	3
10	174311	2021	6
11	174311	2022	7
12	174311	2023	0

Total rows: 72 of 72    Query complete 00:00:00.176

$$T_{Pin\_Code, Year}(\pi_{Pin\_Code, Year, CustomerCount}(\pi_{Pin\_Code, Year} \bowtie COUNT(DISTINCT Customer\_ID) \rightarrow CustomerCount$$

$$(\sigma_{AvgConsumptionPerCustomer < 100}(\pi_{Pin\_Code, Year} \bowtie AVG(YearlyCustomerUsage) \rightarrow AvgConsumptionPerCustomer(Pin\_Code, Year, Customer\_ID$$

$$\bowtie SUM(Total\_Price / Electricity\_rate) \rightarrow YearlyCustomerUsage (Bill \bowtie Meter.Meter\_ID=Bill.Meter\_ID \bowtie Meter \bowtie$$

$$Meter.Customer\_ID=Customer.Customer\_ID \bowtie Bill.rate\_ID=Electricity\_rate.rate\_ID Electricity\_rate))))))$$