

## Task-4

### Procedure:

#### Step-by-step explanation

##### 1. Check existing tables

- We first looked at what tables already exist in the database.
- This helps us avoid naming conflicts and see what data we already have.

##### 2. Export data problem and solution

- When trying to export with COPY, we got a permission error because the database server couldn't write to our local filesystem.
- The fix is to use a client-side export command that writes from your own user account rather than the database server.

##### 3. Create a sales table

- We designed a new table called sales to store coffee sales data, including columns for sale date, coffee type, quantity, price, and total amount.
- This structure allows later analysis with aggregations, filtering, and joins.

##### 4. Insert sample data ○ We added a few rows to simulate real sales transactions. ○

This data lets us test queries before connecting to a real application or importing large datasets.

##### 5. Add a customer relationship ○ We added a customer\_id column so each sale could be linked to a customer. ○ This enables relational database features like joining sales data with customer data.

##### 6. Create a customers table ○ We made a separate table to store customer information such as name and city.

- Keeping customers separate avoids duplication and keeps the data normalized.

##### 7. Update sales to reference customers

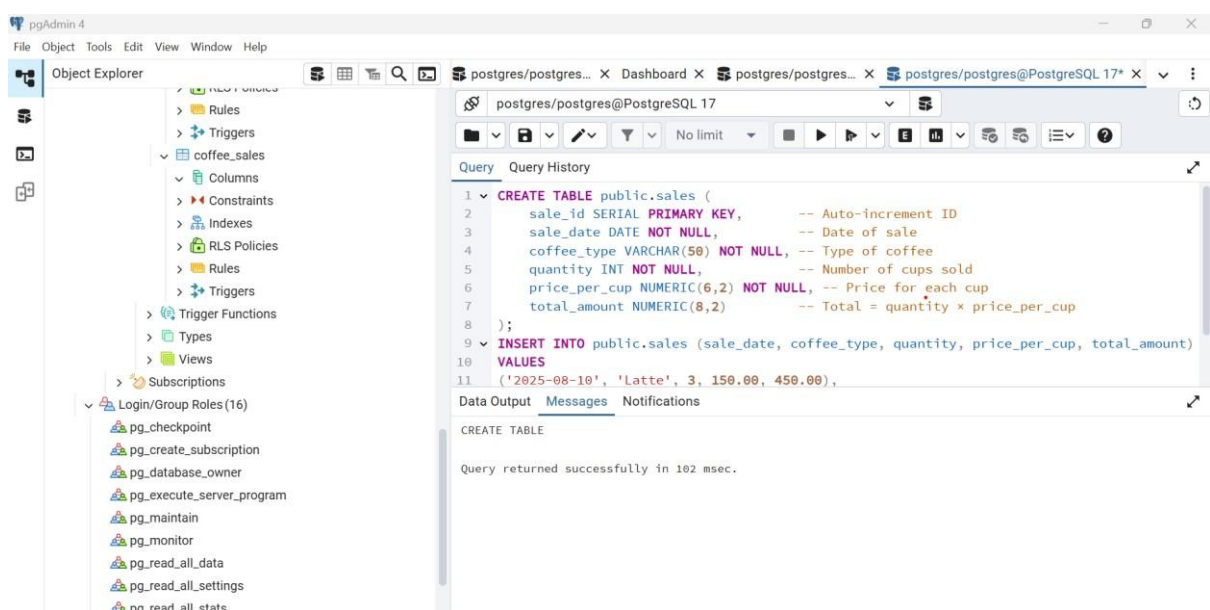
- We linked existing sales to customers by setting the customer\_id value in the sales table.
- Some sales intentionally had no customer to test join behavior.

##### 8. Run join queries ○ INNER JOIN showed only sales with matching customers.

- LEFT JOIN showed all sales whether or not they had a customer.

- **RIGHT JOIN** showed all customers whether or not they made a sale.
9. **Use aggregates and grouping** ○ We calculated total sales and quantities for each coffee type.
- We also found sales above the average value.
10. **Create a view**
- We saved a reusable query as a view so we could easily re-run analysis without retyping the query.
  - This makes reporting faster and cleaner.
11. **Optimize with indexes** ○ We added indexes to speed up queries that filter by coffee type or sale date.
- Indexes improve read performance but slightly slow down writes.
12. **Exporting data**
- We discussed how to export data to CSV using a client command to avoid server permission issues.
  - Choosing a writable folder on your computer avoids most export errors.

## Outputs:



pgAdmin 4

File Object Tools Edit View Window Help

Object Explorer

- Rules
- Triggers
- coffee\_sales
  - Columns
  - Constraints
  - Indexes
  - RLS Policies
  - Rules
  - Triggers
- Trigger Functions
- Types
- Views
- Subscriptions
- Login/Group Roles (16)
  - pg\_checkpoint
  - pg\_create\_subscription
  - pg\_database\_owner
  - pg\_execute\_server\_program
  - pg\_maintain
  - pg\_monitor
  - pg\_read\_all\_data
  - pg\_read\_all\_settings
  - pg\_read\_all\_stats

postgres/postgres@PostgreSQL 17

Query

```
4 coffee_type VARCHAR(50) NOT NULL, -- Type of coffee
5 quantity INT NOT NULL, -- Number of cups sold
6 price_per_cup NUMERIC(6,2) NOT NULL, -- Price for each cup
7 total_amount NUMERIC(8,2) -- Total = quantity * price_per_cup
8 );
9 INSERT INTO public.sales (sale_date, coffee_type, quantity, price_per_cup, total_amount)
10 VALUES
11 ('2025-08-10', 'Latte', 3, 150.00, 450.00),
12 ('2025-08-10', 'Cappuccino', 2, 180.00, 360.00),
13 ('2025-08-10', 'Espresso', 1, 120.00, 120.00);
14
```

Data Output Messages Notifications

INSERT 0 3

Query returned successfully in 70 msec.

postgres/postgres@PostgreSQL 17

Query

```
11 ('2025-08-10', 'Latte', 3, 150.00, 450.00),
12 ('2025-08-10', 'Cappuccino', 2, 180.00, 360.00),
13 ('2025-08-10', 'Espresso', 1, 120.00, 120.00);
14 -- Select all Latte sales in August 2025, sorted by amount (highest first)
15 SELECT *
16 FROM public.sales
17 WHERE coffee_type = 'Latte'
18 AND sale_date BETWEEN '2025-08-01' AND '2025-08-31'
19 ORDER BY total_amount DESC;
20
21 -- Total sales amount by coffee type
```

Data Output Messages Notifications

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

	coffee_type character varying (50)	total_sales numeric
1	Latte	450.00
2	Cappuccino	360.00
3	Espresso	120.00

pgAdmin 4

File Object Tools Edit View Window Help

Object Explorer

- postgres/postgres@PostgreSQL 17
  - Rules
  - Triggers
  - coffee\_sales
    - Columns
    - Constraints
    - Indexes
    - RLS Policies
    - Rules
    - Triggers
  - Trigger Functions
  - Types
  - Views
  - Subscriptions
  - Login/Group Roles (16)
    - pg\_checkpoint
    - pg\_create\_subscription
    - pg\_database\_owner
    - pg\_execute\_server\_program
    - pg\_maintain
    - pg\_monitor
    - pg\_read\_all\_data
    - pg\_read\_all\_settings
    - pg\_read\_all\_stats

postgres/postgres@PostgreSQL 17

Query Query History

```
27 CREATE TABLE public.customers (  
28     customer_id SERIAL PRIMARY KEY,  
29     customer_name VARCHAR(100),  
30     city VARCHAR(50)  
31 );  
32  
33 -- Example customers  
34 INSERT INTO public.customers (customer_name, city)  
35 VALUES  
36 ('Akshaya', 'Chennai'),  
37 ('John', 'Mumbai');
```

Data Output Messages Notifications

INSERT 0 3

Query returned successfully in 115 msec.

pgAdmin 4

File Object Tools Edit View Window Help

Object Explorer

- postgres/postgres@PostgreSQL 17
  - Rules
  - Triggers
  - coffee\_sales
    - Columns
    - Constraints
    - Indexes
    - RLS Policies
    - Rules
    - Triggers
  - Trigger Functions
  - Types
  - Views
  - Subscriptions
  - Login/Group Roles (16)
    - pg\_checkpoint
    - pg\_create\_subscription
    - pg\_database\_owner
    - pg\_execute\_server\_program
    - pg\_maintain
    - pg\_monitor
    - pg\_read\_all\_data
    - pg\_read\_all\_settings
    - pg\_read\_all\_stats

postgres/postgres@PostgreSQL 17

Query Query History

```
32  
33 -- Example customers  
34 INSERT INTO public.customers (customer_name, city)  
35 VALUES  
36 ('Akshaya', 'Chennai'),  
37 ('John', 'Mumbai'),  
38 ('Priya', 'Delhi');  
39  
40 ALTER TABLE public.sales  
41 ADD COLUMN customer_id INT;  
42
```

Data Output Messages Notifications

ALTER TABLE

Query returned successfully in 68 msec.

pgAdmin 4

File Object Tools Edit View Window Help

Object Explorer

- postgres/postgres@PostgreSQL 17
  - Rules
  - Triggers
  - coffee\_sales
    - Columns
    - Constraints
    - Indexes
    - RLS Policies
    - Rules
    - Triggers
  - Trigger Functions
  - Types
  - Views
  - Subscriptions
- Login/Group Roles (16)
  - pg\_checkpoint
  - pg\_create\_subscription
  - pg\_database\_owner
  - pg\_execute\_server\_program
  - pg\_maintain
  - pg\_monitor
  - pg\_read\_all\_data
  - pg\_read\_all\_settings
  - pg\_read\_all\_stats

postgres/postgres@PostgreSQL 17

Query Query History

```
34 INSERT INTO public.customers (customer_name, city)
35 VALUES
36 ('Akshaya', 'Chennai'),
37 ('John', 'Mumbai'),
38 ('Priya', 'Delhi');
39
40 ALTER TABLE public.sales
41 ADD COLUMN customer_id INT
42 UPDATE public.sales
43 SET customer_id = 1
44 WHERE sale_id = 1;
```

Data Output Messages Notifications

UPDATE 1

Query returned successfully in 68 msec.

pgAdmin 4

File Object Tools Edit View Window Help

Object Explorer

- postgres/postgres@PostgreSQL 17
  - Rules
  - Triggers
  - coffee\_sales
    - Columns
    - Constraints
    - Indexes
    - RLS Policies
    - Rules
    - Triggers
  - Trigger Functions
  - Types
  - Views
  - Subscriptions
- Login/Group Roles (16)
  - pg\_checkpoint
  - pg\_create\_subscription
  - pg\_database\_owner
  - pg\_execute\_server\_program
  - pg\_maintain
  - pg\_monitor
  - pg\_read\_all\_data
  - pg\_read\_all\_settings
  - pg\_read\_all\_stats
  - pg\_read\_server\_files

postgres/postgres@PostgreSQL 17

Query Query History

```
65 SELECT s.sale_id, s.coffee_type, c.customer_name
66 FROM public.sales s
67 LEFT JOIN public.customers c
68 ON s.customer_id = c.customer_id;
69
70 -- RIGHT JOIN: all customers, even those with no sales
71 SELECT s.sale_id, s.coffee_type, c.customer_name
72 FROM public.sales s
73 RIGHT JOIN public.customers c
74 ON s.customer_id = c.customer_id;
75
```

Data Output Messages Notifications

Showing rows: 1 to 4 Page No: 1 of 1

sale_id	coffee_type	customer_name
1	Latte	Akshaya
2	Cappuccino	John
3	Espresso	Akshaya
4	[null]	Priya

Total rows: 4 Query complete 00:00:00.067 CRLF Ln 75, Col 1

pgAdmin 4

File Object Tools Edit View Window Help

Object Explorer

- postgres/postgres@PostgreSQL 17
  - Rules
  - Triggers
  - coffee\_sales
    - Columns
    - Constraints
    - Indexes
    - RLS Policies
    - Rules
    - Triggers
  - Trigger Functions
  - Types
  - Views
  - Subscriptions
- Login/Group Roles (16)
  - pg\_checkpoint
  - pg\_create\_subscription

postgres/postgres@PostgreSQL 17

Data Output Messages Notifications

Showing rows: 1 to 8 Page No: 1 of 1

sale_id	sale_date	coffee_type	quantity	price_per_cup	total_amount	customer_id
1	2025-08-10	Latte	3	150.00	450.00	1
2	2025-08-10	Cappuccino	2	180.00	360.00	2
3	2025-08-10	Espresso	1	120.00	120.00	1
4	2025-08-10	Latte	2	150.00	300.00	1
5	2025-08-10	Cappuccino	1	180.00	180.00	1
6	2025-08-10	Espresso	3	120.00	360.00	2
7	2025-08-10	Mocha	4	200.00	800.00	3
8	2025-08-10	Americano	2	100.00	200.00	[null]

