Worksheet 2

Student Name: Aryan **UID: 25MCI10082**
Branch: MCA (AI&ML) **Section/Group: 1/A**
Semester: 2nd **Date of Performance: 22/01/2026**
Subject Name:- DBMS LAB **Subject Code:**

1. Aim of the Session

To implement and analyze SQL SELECT queries using filtering, sorting, grouping, and aggregation concepts in PostgreSQL for efficient data retrieval and analytical reporting.

2. Objective of the Session

- To retrieve specific data using filtering conditions
- To sort query results using single and multiple attributes
- To perform aggregation using grouping techniques
- To apply conditions on aggregated data
- To understand real-world analytical queries commonly asked in placement interviews

3. Practical / Experiment Steps

Design the database schema for Customer Orders.

Create table using appropriate constraints.

Insert sample records into tables.

Perform filtering query.

Perform Sorting query.

Perform Grouping and Aggregation.

Use Having clause.

4. Procedure of the Practical

(i) Start the system and log in to the computer.

(ii) Open PostgreSQL software.

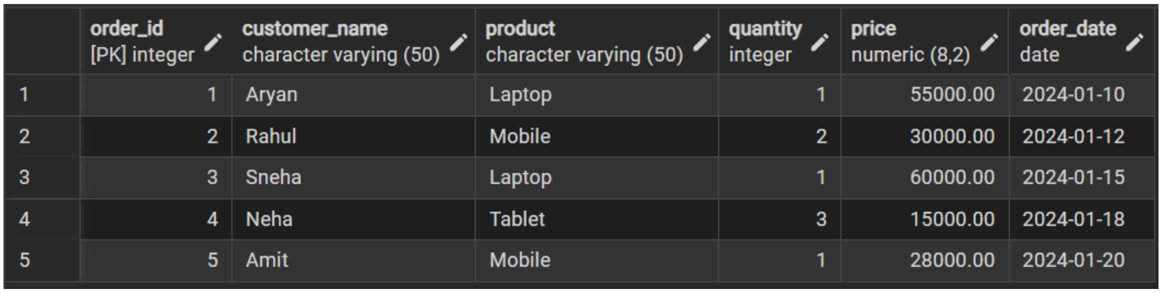
(iii) create database Experiment2;

(iv) Create a table for customer orders.

```
CREATE TABLE customer_orders (
    order_id SERIAL PRIMARY KEY,
    customer_name VARCHAR(50),
    product VARCHAR(50),
    quantity INT,
    price NUMERIC(8,2),
    order_date DATE
);
```

(v) Insert records using DML commands.

```
INSERT INTO customer_orders
(customer_name, product, quantity, price, order_date)
VALUES
('Aryan', 'Laptop', 1, 55000, '2024-01-10'),
('Rahul', 'Mobile', 2, 30000, '2024-01-12'),
('Sneha', 'Laptop', 1, 60000, '2024-01-15'),
('Neha', 'Tablet', 3, 15000, '2024-01-18'),
('Amit', 'Mobile', 1, 28000, '2024-01-20');
```

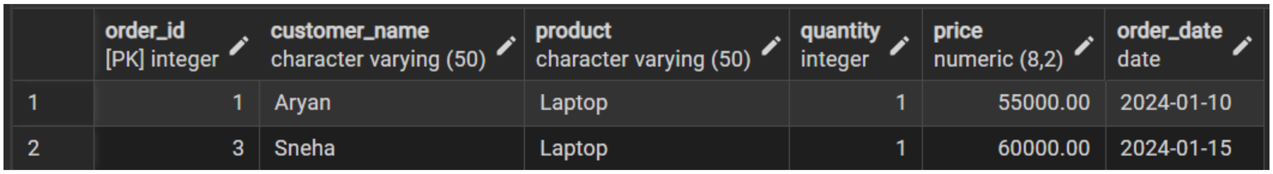


	order_id [PK] integer	customer_name character varying (50)	product character varying (50)	quantity integer	price numeric (8,2)	order_date date
1	1	Aryan	Laptop	1	55000.00	2024-01-10
2	2	Rahul	Mobile	2	30000.00	2024-01-12
3	3	Sneha	Laptop	1	60000.00	2024-01-15
4	4	Neha	Tablet	3	15000.00	2024-01-18
5	5	Amit	Mobile	1	28000.00	2024-01-20

(vi) Filtering Query.

SELECT * FROM customer_orders

WHERE price > 30000;

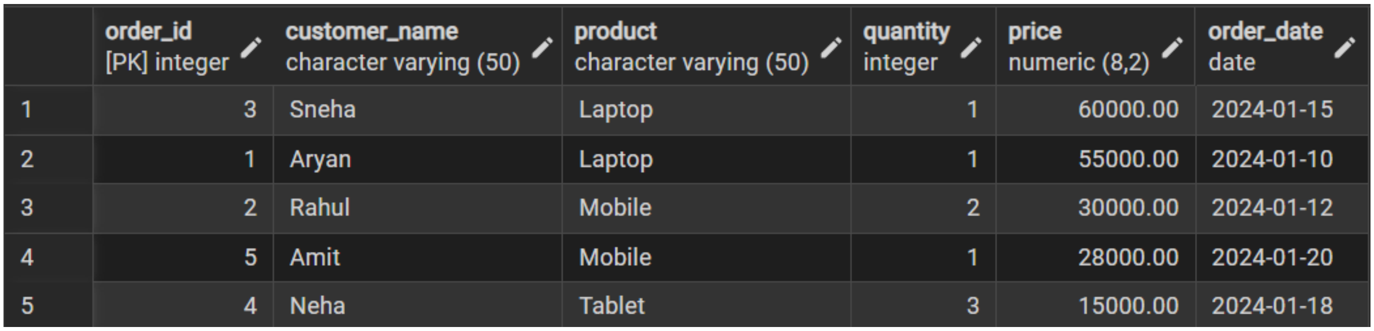


	order_id [PK] integer	customer_name character varying (50)	product character varying (50)	quantity integer	price numeric (8,2)	order_date date
1	1	Aryan	Laptop	1	55000.00	2024-01-10
2	3	Sneha	Laptop	1	60000.00	2024-01-15

(vii) Sorting query

SELECT * FROM customer_orders

ORDER BY price DESC;



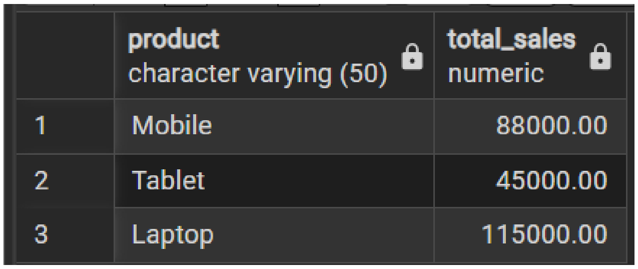
	order_id [PK] integer	customer_name character varying (50)	product character varying (50)	quantity integer	price numeric (8,2)	order_date date
1	3	Sneha	Laptop	1	60000.00	2024-01-15
2	1	Aryan	Laptop	1	55000.00	2024-01-10
3	2	Rahul	Mobile	2	30000.00	2024-01-12
4	5	Amit	Mobile	1	28000.00	2024-01-20
5	4	Neha	Tablet	3	15000.00	2024-01-18

(viii) Grouping and Aggregation.

SELECT product, SUM(price * quantity) AS total_sales

FROM customer_orders

GROUP BY product;



	product character varying (50)	total_sales numeric
1	Mobile	88000.00
2	Tablet	45000.00
3	Laptop	115000.00

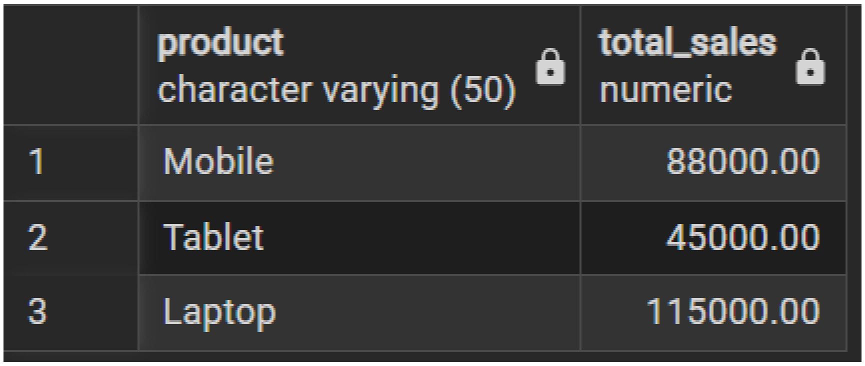
(ix) HAVING Clause.

SELECT product, SUM(price * quantity) AS total_sales

FROM customer_orders

GROUP BY product

HAVING SUM(price * quantity) > 30000;



	product character varying (50)	total_sales numeric
1	Mobile	88000.00
2	Tablet	45000.00
3	Laptop	115000.00

Learning Outcomes

- How data can be filtered to retrieve only relevant records from a database.
- Learn how sorting improves readability and usefulness of query results in reports.
- Gain the ability to group data for analytical purposes.
- Differentiate between row-level conditions and group-level conditions.
- Develop confidence in writing analytical SQL queries used in real-world scenarios.
- Better prepared to answer SQL-based placement and interview questions related to filtering, grouping, and aggregation.