Worksheet No. 2

Student Name: Ashi Gupta
UID: 25MCA20160
Branch: MCA Gen
Section/Group: MCA-1_A
Semester: II
Date of Performance: 19/01/2026

Subject Name: Technical Training
Subject Code: 25CAP-652

1.Aim/Overview of the practical:

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

2. Objectives:

- 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

S/W Requirement: Oracle Database Express Edition and pgAdmin

Practical:

Step 1: Database and Table Preparation

```
create table orders (
order_id serial primary key,
customer_name varchar(50),
product varchar(50),
quantity int,
```

price numeric(10,2),

order_date date

);

insert into orders (customer_name, product, quantity, price, order_date) values

('amit', 'laptop', 1, 65000, '2024-01-10'),

('neha', 'mobile', 2, 40000, '2024-01-12'),

('rohan', 'tablet', 1, 25000, '2024-01-15'),

('simran', 'laptop', 1, 70000, '2024-01-18'),

('ankit', 'mobile', 3, 60000, '2024-01-20'),

('pooja', 'headphones', 2, 5000, '2024-01-22'),

('rahul', 'tablet', 2, 48000, '2024-01-25');

Step 2: Filtering Data Using Conditions

select * from orders where price > 50000;

	order_id [PK] integer	customer_name character varying (50)	product character varying (50)	quantity integer	price numeric (10,2)	order_date date
1	1	amit	laptop	1	65000.00	2024-01-10
2	4	simran	laptop	1	70000.00	2024-01-18
3	5	ankit	mobile	3	60000.00	2024-01-20

select customer_name, product, price from orders where price > 30000 and quantity >= 2;

	customer_name character varying (50)	product character varying (50)	price numeric (10,2)
1	neha	mobile	40000.00
2	ankit	mobile	60000.00
3	rahul	tablet	48000.00

Step 3: Sorting Query Results

select customer_name, product, price from orders order by price asc;

	customer_name character varying (50)	product character varying (50)	price numeric (10,2)
1	pooja	headphones	5000.00
2	rohan	tablet	25000.00
3	neha	mobile	40000.00
4	rahul	tablet	48000.00
5	ankit	mobile	60000.00
6	amit	laptop	65000.00
7	simran	laptop	70000.00

select customer_name, product, price from orders order by price desc;

	customer_name character varying (50)	product character varying (50)	price numeric (10,2)
1	simran	laptop	70000.00
2	amit	laptop	65000.00
3	ankit	mobile	60000.00
4	rahul	tablet	48000.00
5	neha	mobile	40000.00
6	rohan	tablet	25000.00
7	pooja	headphones	5000.00

select customer_name, product, price, quantity from orders order by product asc, price desc;

	customer_name character varying (50)	product character varying (50)	price numeric (10,2)	quantity integer
1	pooja	headphones	5000.00	2
2	simran	laptop	70000.00	1
3	amit	laptop	65000.00	1
4	ankit	mobile	60000.00	3
5	neha	mobile	40000.00	2
6	rahul	tablet	48000.00	2
7	rohan	tablet	25000.00	1

Step 4: Grouping Data for Aggregation

select product, sum(price) as total_sales from orders group by product;

	product character varying (50)	total_sales numeric
1	headphones	5000.00
2	laptop	135000.00
3	tablet	73000.00
4	mobile	100000.00

select product, sum(quantity) as total_quantity from orders group by product;

	product character varying (50)	total_quantity bigint
1	headphones	2
2	laptop	2
3	tablet	3
4	mobile	5

Step 5: Applying Conditions on Aggregated Data

select product, sum(price) as total_sales from orders group by product having sum(price) > 50000;

	product character varying (50)	total_sales numeric
1	laptop	135000.00
2	tablet	73000.00
3	mobile	100000.00

Step 6: Conceptual Understanding of Filtering vs Aggregation Conditions

select product, sum(price) from orders where sum(price) > 50000 group by product;

```
ERROR:  aggregate functions are not allowed in WHERE
LINE 1: select product, sum(price) from orders where sum(price) > 50...
```

select product, sum(price) from orders group by product having sum(price) > 50000;

	product character varying (50)	sum numeric
1	laptop	135000.00
2	tablet	73000.00
3	mobile	100000.00

Learning Outcomes

● Understand how conditional filtering is used to retrieve only relevant records from a database.

● Explain how sorting enhances the readability and usefulness of query results in reports.

● Apply grouping techniques to organize data for analytical and summary purposes.

● Distinguish clearly between row-level conditions and group-level conditions using appropriate sql clauses.

● Develop confidence in writing analytical sql queries applicable to real-world database scenarios.

● Demonstrate improved readiness for placement and interview questions related to filtering, grouping, and aggregation concepts.