Student Name: Husanpreet Kaur **UID: 24BCY70138**
Branch: AIT-CSE (Cyber Security) **Section/Group: 24AIT-KRG2**
Semester: 4 **Date of Performance:16/01/2026**
Subject Name: Database Management System **Subject Code: 24CSH-298**

WORKSHEET 2

AIM: To design and implement a sample database system using DDL, DML, and DCL commands, including database creation, data manipulation, schema modification, and role-based access control to ensure data integrity and secure, read-only access for authorized users.

S/W Requirement: Oracle Database Express Edition and pgAdmin

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

Given:

Practical / Experiment Steps

Step 1: Database and Table Preparation

- Start the PostgreSQL server.
- Open the PostgreSQL client tool.
- Create a database for the experiment.
- Prepare a sample table representing customer orders containing details such as customer name, product, quantity, price, and order date.
- Insert sufficient sample records to allow meaningful analysis.

Query:

Table Creation:

```
CREATE TABLE Students1 (
   id NUMERIC PRIMARY KEY,
   name VARCHAR(50),
   city VARCHAR(30),
```

```
    marks NUMERIC(10,0)
);
```

Insertion Of Records

```
INSERT INTO Students1 VALUES (1, 'Aman', 'Mohali', 85);
INSERT INTO Students1 VALUES (2, 'Rohit', 'Mohali', 78);
INSERT INTO Students1 VALUES (3, 'Neha', 'Mohali', 92);
INSERT INTO Students1 VALUES (4, 'Simran', 'Amritsar', 88);
INSERT INTO Students1 VALUES (6, 'Karan', 'Amritsar', 75);
SELECT * FROM Students1;
```

Output:

Data Output | Messages | Notifications

	id [PK] numeric	name character varying (50)	city character varying (30)	marks numeric (10)
1	5	Karan	Amritsar	75
2	1	Aman	Mohali	85
3	2	Rohit	Mohali	78
4	3	Neha	Mohali	92
5	4	Simran	Amritsar	88
6	6	Karan	Amritsar	75

Step 2: Filtering Data Using Conditions

Query:

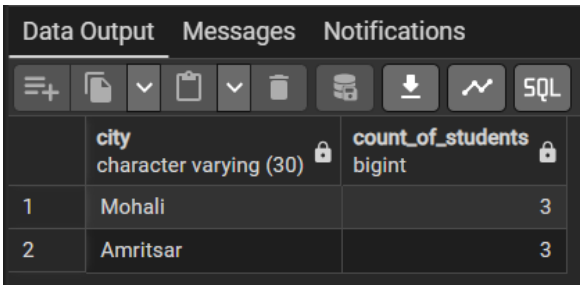
```
SELECT city,count(*) as count_of_students
from Students1
group by city;

SELECT city,count(id) as count_of_students
from Students1
group by city;

SELECT city,COUNT(ID) as count_of_students
FROM Students1
GROUP BY city
```

HAVING COUNT(ID)>=3;

Output:



Step 3: Sorting Query Results

- Retrieve selected columns from the table and arrange the output based on numerical values such as price.
- Perform sorting using both ascending and descending order.
- Apply sorting on more than one attribute to understand priority-based ordering.

Query:

Sort orders by price in ascending order

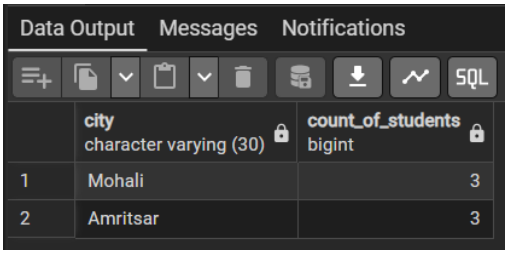
SELECT city, COUNT(*) as count_of_students

FROM Students1

GROUP BY city

ORDER BY COUNT(*) ASC;

Output:



Step 4: Grouping Data for Aggregation

Query:

SELECT city,AVG(MARKS)::NUMERIC(10,2) as average_marks

FROM Students1

Group BY city;

Output:

Data Output | Messages | Notifications

	city character varying (30)	average_marks numeric (10,2)
1	Mohali	85.00
2	Amritsar	79.33

SELECT city, SUM(marks) FROM Students1 GROUP BY city;

Data Output | Messages | Notifications

	city character varying (30)	sum numeric
1	Mohali	255
2	Amritsar	238

SELECT city, MAX(marks) FROM Students1 GROUP BY city;

Data Output | Messages | Notifications

	city character varying (30)	max numeric
1	Mohali	92
2	Amritsar	88

SELECT city, MIN(marks) FROM Students1 GROUP BY city;

Data Output | Messages | Notifications

	city character varying (30)	min numeric
1	Mohali	78
2	Amritsar	75

Learning Outcomes:

- Understand how to create relational database tables using appropriate data types and constraints
- Learn to retrieve required data from a table using **row-level filtering** with the WHERE clause.
- Gain the ability to apply **column-level (group-level) filtering** using the HAVING clause.
- Understand the use of **aggregate functions** such as SUM(), AVG(), and COUNT() for analytical reporting.
- Clearly differentiate between **row-level filtering and group-level filtering**, and apply them correctly in real-world SQL scenarios.


CU
CHANDIGARH UNIVERSITY
CHANDIGARH
UNIVERSITY
Discover. Learn. Empower.


NAAC
GRADE A+