Experiment 2:-

Student Name: Mohd Rehan Khan UID: __24bai70004_____
Branch: CSE - AIML Section/Group: _24_AIT_KRG-G1
Semester: 4_ Date of Performance: 16/01/2026
Subject Name: DBMS Subject Code: 24CSH-298

Aim

To understand and implement SQL SELECT queries using WHERE, GROUP BY, HAVING, and ORDER BY clauses for retrieving and analyzing data from relational database tables.

Software Requirements

Database Management System:
PostgreSQL / Oracle XE

Database Administration Tool: pgAdmin

Objectives

- To practice SQL SELECT queries
- To apply filtering using WHERE clause
- To group records using GROUP BY
- To apply conditions using HAVING
- To sort records using ORDER BY
- To understand aggregate functions

Practical / Experiment Steps

1. Create EMPLOYEE table
2. Insert sample employee records
3. Verify table data
4. Apply salary condition
5. Perform grouping and aggregation
6. Filter grouped data
7. Sort the final output

Procedure

(i) Start the system and login.
(ii) Open pgAdmin / Oracle Database.
(iii) Create or select the database.
(iv) Create EMPLOYEE table.
(v) Insert records into EMPLOYEE table.
(vi) Execute SELECT queries step by step.
(vii) Verify the output.
(viii) Save work and capture screenshots.

Detailed SQL Code

```
-- Step 1: Create EMPLOYEE table
CREATE TABLE employee (
emp_id NUMBER PRIMARY KEY,
emp_name VARCHAR2(50),
department VARCHAR2(30),
salary NUMBER,    joining_date
DATE
);

-- Step 2: Insert records
INSERT INTO employee VALUES (101, 'Aman', 'IT', 50000, '01-JAN-2023');
INSERT INTO employee VALUES (102, 'Rahul', 'HR', 25000, '12-MAR-2022');
INSERT INTO employee VALUES (103, 'Neha', 'IT', 60000, '05-JUN-2021');
INSERT INTO employee VALUES (104, 'Ravi', 'Finance', 40000, '18-AUG-2020');
INSERT INTO employee VALUES (105, 'Priya', 'HR', 35000, '10-FEB-2023');

COMMIT;

-- Step 3: Display all records
SELECT * FROM employee;

-- Step 4: Filter employees with salary > 20000
SELECT * FROM employee WHERE salary > 20000;

-- Step 5: Calculate average salary department-wise
SELECT department, AVG(salary) avg_salary
FROM employee
GROUP BY department;

-- Step 6: Apply HAVING clause
SELECT department, AVG(salary) avg_salary
FROM employee
GROUP BY department
HAVING AVG(salary) > 30000;

-- Step 7: Final required query
SELECT department, AVG(salary) avg_salary
FROM employee
WHERE salary > 20000
GROUP BY department
HAVING AVG(salary) > 30000
ORDER BY avg_salary DESC;
```

Input / Output Details

Input:
Employee records and salary condition (>20000)

Output:

Table Created:-

```
CREATE TABLE

Query returned successfully in 101 msec.
```

Insert Values:-

```
INSERT 0 1

Query returned successfully in 121 msec.
```

	emp_id [PK] integer	emp_name character varying (50)	department character varying (30)	salary integer	joining_date date
1	101	Aman	IT	50000	2023-01-01
2	102	Rahul	HR	25000	2022-03-12
3	103	Neha	IT	60000	2021-06-05
4	104	Ravi	Finance	40000	2020-08-18
5	105	Priya	HR	35000	2023-02-10

Filter employees with salary > 20000

	emp_id [PK] integer	emp_name character varying (50)	department character varying (30)	salary integer	joining_date date
1	101	Aman	IT	50000	2023-01-01
2	102	Rahul	HR	25000	2022-03-12
3	103	Neha	IT	60000	2021-06-05
4	104	Ravi	Finance	40000	2020-08-18
5	105	Priya	HR	35000	2023-02-10

	department character varying (30)	avg_salary numeric
1	Finance	40000.000000000000
2	IT	55000.000000000000
3	HR	30000.000000000000

	department character varying (30)	avg_salary numeric
1	Finance	40000.000000000000
2	IT	55000.000000000000

Learning Outcomes

- Learned to write SQL SELECT queries
- Understood WHERE, GROUP BY, HAVING, ORDER BY
- Gained hands-on experience in database querying