

Student Name: Punya Arora

UID:24BAI70928

Branch: Computer Science & Engineering (AIML)

Section/Group:24AIT-KRG1/G2

Semester:4th

Subject Name: DBMS

1. Aim

To understand and implement SQL SELECT queries using various clauses such as WHERE, ORDER BY, GROUP BY, and HAVING to retrieve and manipulate data efficiently from relational database tables.

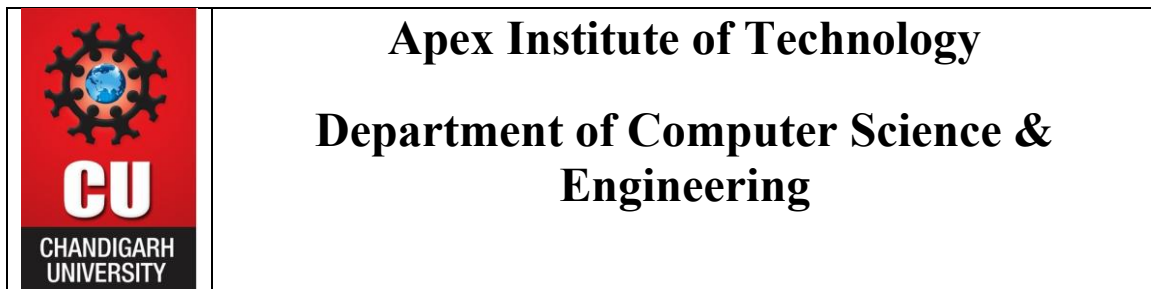
2. Objective of the Session

- To practice writing SQL SELECT statements.
- To apply filtering conditions using the WHERE clause.
- To sort query results using the ORDER BY clause.
- To group records using the GROUP BY clause.
- To filter grouped data using the HAVING clause.
- To analyze data using aggregate functions like COUNT(), SUM(), AVG(), MIN(), and MAX().

3. Practical / Experiment Steps

1. Display the department name and the average salary of employees for each department.
2. Consider only those employees whose salary is greater than 20,000.
3. Display only those departments where the average salary is greater than 30,000.
4. Arrange the final output in descending order of average salary.

4. Procedure of the Practical



- (1) Start the system and log in to the computer.
- (2) Open PgAdmin (PostgreSQL).
- (3) Create or select the required database (e.g., lab_db).
- (4) Create the EMPLOYEE table using the given schema.
- (5) Insert sample data into the EMPLOYEE table.
- (6) Execute the queries step-by-step according to the practical steps.
- (7) Verify the output after each query execution.
- (8) Capture screenshots of execution and results for record.
- (9) Save the work and upload worksheet (Word + PDF) on GitHub.

5. I/O Analysis (Input / Output Analysis)

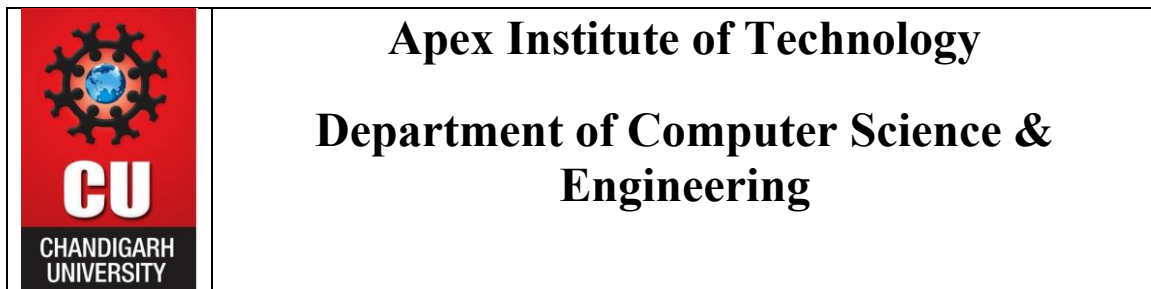
Input: SQL commands and queries executed in PgAdmin (table creation, insertion, and SELECT queries).

Output: Result tables displayed in PgAdmin showing department-wise average salary after applying WHERE, HAVING, and ORDER BY clauses.

SQL Implementation (PgAdmin / PostgreSQL)

DROP TABLE IF EXISTS employee;

**CREATE TABLE employee(
 emp_id NUMERIC PRIMARY KEY,
 emp_name VARCHAR(50),
 department VARCHAR(50),**



```
salary NUMERIC(10,2)

);

INSERT INTO employee VALUES

(725, 'Punya Arora', 'Sales', 25000.00),

(131, 'Arnav Prajapati', 'HR', 32000.00),

(732, 'Akshveer Singh Lamba', 'Management', 35000.00),

(735, 'Anuj Kumar', 'Accounts', 30000.00);

SELECT * FROM employee;

SELECT department, AVG(salary) AS avg_salary

FROM employee

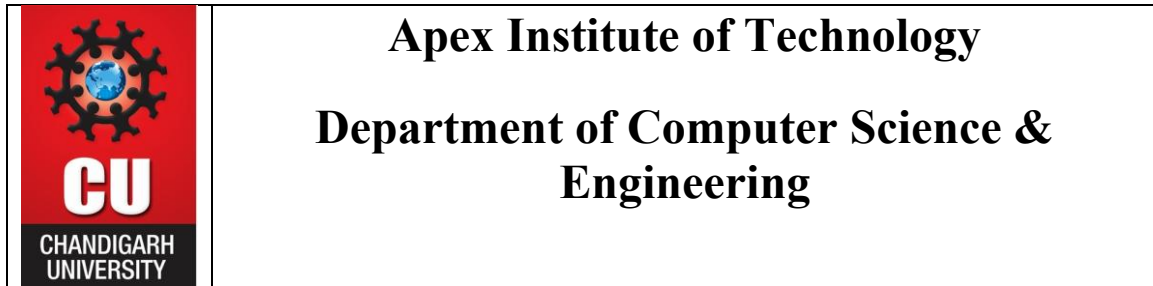
GROUP BY department;

SELECT department, AVG(salary) AS avg_salary

FROM employee

GROUP BY department

HAVING AVG(salary) > 20000.00
```



SELECT department , AVG(salary) AS avg_salary

FROM employee

WHERE salary > 30000.00

GROUP BY department

HAVING AVG(salary) > 20000.00

ORDER BY avg_salary DESC

6. Learning Outcome

- Understood the syntax and usage of SQL SELECT statements.
- Gained practical knowledge of WHERE clause for filtering rows.
- Learned grouping operations using GROUP BY clause.
- Applied HAVING clause to filter grouped results.
- Sorted query outputs using ORDER BY clause.
- Got hands-on experience in PostgreSQL execution using PgAdmin.

Screenshots:

	emp_id [PK] numeric	emp_name character varying (50)	department character varying (50)	salary numeric (10,2)
1	725	Punya Arora	Sales	25000.00
2	131	Arnav Prajapati	HR	32000.00
3	732	Akshveer Singh Lamba	Management	35000.00
4	735	Anuj Kumar	Accounts	30000.00

Data Output Messages Notifications		
<div> <div>≡ +</div> <div> <div>📄</div> <div>▼</div> <div>📋</div> <div>▼</div> <div>🗑</div> <div>🗄</div> <div>⬇</div> <div>📈</div> <div>SQL</div> </div> </div>		
	department character varying (50)	avg_salary numeric
1	Accounts	30000.000000000000
2	Management	35000.000000000000
3	Sales	25000.000000000000
4	HR	32000.000000000000

Data Output Messages Notifications		
<div> <div>≡ +</div> <div> <div>📄</div> <div>▼</div> <div>📋</div> <div>▼</div> <div>🗑</div> <div>🗄</div> <div>⬇</div> <div>📈</div> <div>SQL</div> </div> </div>		
	department character varying (50)	avg_salary numeric
1	Management	35000.000000000000
2	HR	32000.000000000000