

School of Computer Science, UPES, Dehradun.

A

LABORATORY FILE

On

DATABASE MANAGEMENT SYSTEM (DBMS) LAB

B.TECH. -III Semester

AUG. – NOV.- 2024.

Submitted by:

Name: Harsh Vardhan Saini SAP ID: 500120369 Roll No: R2142230056

Batch: 2

DBMS Lab

B.Tech. III Sem.

Experiment 08

Use of Inbuilt functions and relational algebra operation

Aim:

To create a relational database that stores information about departments and employees, and to perform various SQL queries to analyze employee data.

Problem Statement:

The task is to manage employee and department data effectively, allowing for various analyses such as average salary calculation, employee counts, and salary comparisons across different roles and departments. The goal is to facilitate easy retrieval of information for decision-making and reporting.

Theory:

Relational databases use tables to store data in a structured format, allowing for relationships to be established between different entities (in this case, employees and departments). SQL (Structured Query Language) is used to query and manipulate the data stored in these tables. Key concepts include:

- Tables: A way to organize data into rows and columns.
- Primary Keys: A unique identifier for each record in a table.
- Foreign Keys: A field in one table that uniquely identifies a row of another table, establishing a relationship between them.
- Aggregate Functions: Functions such as `AVG()`, `SUM()`, and `COUNT()` used to perform calculations on a set of values.

Commands Used:

- 1. Database and Table Creation
- 2. Data Insertion

DBMS Lab

B.Tech. III Sem.

3. Data Retrieval Queries:

Results:

```
-- AYUSH VASHISHTH
        -- 500119331
      CREATE DATABASE exp8;
       USE exp8;
7 • 

CREATE TABLE DEPT (
8
       DEPTNO INT PRIMARY KEY,
9
       DNAME VARCHAR(20),
       LOC VARCHAR(20)
10
     );
11
12 • 

CREATE TABLE EMP (
      EMPNO INT PRIMARY KEY,
      ENAME VARCHAR(20),
      JOB VARCHAR(20),
15
      MGR INT,
16
      HIREDATE DATE,
17
      SAL DECIMAL(10,2),
18
19
       COMM DECIMAL(10,2),
       DEPTNO INT,
       FOREIGN KEY (DEPTNO) REFERENCES DEPT(DEPTNO)
21
     );
22
23
      INSERT INTO DEPT (DEPTNO, DNAME, LOC) VALUES (10, 'ACCOUNTING', 'NEW YORK');
24
25 •
      INSERT INTO DEPT (DEPTNO, DNAME, LOC) VALUES (20, 'RESEARCH', 'DALLAS');
      INSERT INTO DEPT (DEPTNO, DNAME, LOC) VALUES (30, 'SALES', 'CHICAGO');
     INSERT INTO DEPT (DEPTNO, DNAME, LOC) VALUES (40, 'OPERATIONS', 'BOSTON');
```

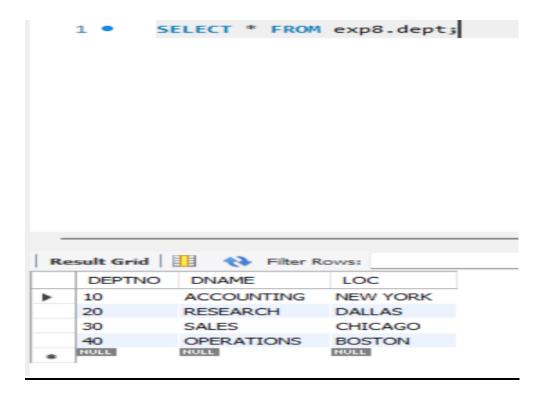
DBMS Lab B.Tech. III Sem.

```
INSERT INTO EMP (EMPNO, ENAME, JOB, MGR, HIREDATE, SAL, COMM, DEPTNO) VALUES (7369, 'SMITH', 'CLERK', 7902, '1980-12-17', 500, 800, 20);
       INSERT INTO EMP (EMPNO, ENAME, JOB, MGR, HIREDATE, SAL, COMM, DEPTNO) VALUES (7499, 'ALLEN', 'SALESMAN', 7698, '1981-02-20', 1600, 300, 300);
30 0
31 •
       INSERT INTO EMP (EMPNO, ENAME, JOB, MGR, HIREDATE, SAL, COMM, DEPTNO) VALUES (7521, 'WARD', 'SALESMAN', 7698, '1981-02-22', 1250, 500, 30);
        INSERT INTO EMP (EMPNO, ENAME, JOB, MGR, HIREDATE, SAL, COMM, DEPTNO) VALUES (7566, 'JONES', 'MANAGER', 7839, '1981-04-02', 2975, NULL, 20);
33 •
       INSERT INTO EMP (EMPNO, ENAME, JOB, MGR, HIREDATE, SAL, COMM, DEPTNO) VALUES (7654, 'MARTIN', 'SALESMAN', 7698, '1981-09-28', 1250, 1400, 30);
       INSERT INTO EMP (EMPNO, ENAME, JOB, MGR, HIREDATE, SAL, COMM, DEPTNO) VALUES (7698, 'BLAKE', 'MANAGER', 7839, '1981-05-01', 2850, NULL, 30);
34 •
35 • INSERT INTO EMP (EMPNO, ENAME, JOB, MGR, HIREDATE, SAL, COMM, DEPTNO) VALUES (7782, 'CLARK', 'MANAGER', 7839, '1981-06-09', 2450, NULL, 10);
36 INSERT INTO EMP (EMPNO, ENAME, JOB, MGR, HIREDATE, SAL, COMM, DEPTNO) VALUES (7788, 'SCOTT', 'ANALYST', 7566, '1982-12-09', 3000, NULL, 20);
       INSERT INTO EMP (EMPNO, ENAME, JOB, HIREDATE, SAL, COMM, DEPTNO) VALUES (7839, 'KING', 'PRESIDENT', '1981-11-17', 5000, NULL, 10);
37 •
       INSERT INTO EMP (EMPNO, ENAME, JOB, MGR, HIREDATE, SAL, COMM, DEPTNO) VALUES (7844, 'TURNER', 'SALESMAN', 7698, '1981-09-08', 1500, 0, 30);
38 •
       INSERT INTO EMP (EMPNO, ENAME, JOB, MGR, HIREDATE, SAL, COMM, DEPTNO) VALUES (7876, 'ADAMS', 'CLERK', 7788, '1983-01-12', 1100, NULL, 20);
39 •
40 •
       INSERT INTO EMP (EMPNO, ENAME, JOB, MGR, HIREDATE, SAL, COMM, DEPTNO) VALUES (7900, 'JAMES', 'CLERK', 7698, '1981-12-03', 950, NULL, 30);
        INSERT INTO EMP (EMPNO, ENAME, JOB, MGR, HIREDATE, SAL, COMM, DEPTNO) VALUES (7902, 'FORD', 'ANALYST', 7566, '1981-12-03', 3000, NULL, 20);
41 •
        INSERT INTO EMP (EMPNO, ENAME, JOB, MGR, HIREDATE, SAL, COMM, DEPTNO) VALUES (7934, 'MILLER', 'CLERK', 7782, '1982-01-23', 1300, NULL, 10);
42 •
43
        -- Retrieve the average salary of all employees
44
45 0
       SELECT AVG(SAL) AS Avg Salary
46
        FROM EMP:
47
         -- Retrieve the number of employees
48 •
        SELECT COUNT(*) AS Total_Employees
49
         FROM EMP:
         -- Retrieve distinct number of employee
50
51 •
        SELECT COUNT(DISTINCT ENAME) AS Distinct_Employees
52
53
         -- Retrieve total salary of employee group by job
54 •
        SELECT JOB, SUM(SAL) AS Total_Salary
         FROM EMP
55
         GROUP BY JOB:
```

```
-- Display the employee information with maximum salary
57
         SELECT *
58 0
         FROM EMP
59
         WHERE SAL = (SELECT MAX(SAL) FROM EMP);
68
         -- Find the highest paid employee in department 10.
61
62 •
        SELECT *
         FROM EMP
63
     WHERE SAL = (SELECT MAX(SAL) FROM EMP WHERE
64
      DEPTNO = 10);
        -- List the emps whose sal is equal to the average of max and minimum
66
67 •
        SELECT *
68
        FROM EMP
         WHERE SAL = (SELECT (MAX(SAL) + MIN(SAL)) / 2 FROM EMP);
69
         -- List the emps who joined in the company on the same date.
71 •
        SELECT ENAME, HIREDATE
         FROM EMP
72
        GROUP BY HIREDATE, ENAME
73
        HAVING COUNT(HIREDATE) > 1;
74
75
         -- Display the employee names in upper and lower case
         SELECT UPPER(ENAME) AS Upper Case Name,
76 •
        LOWER(ENAME) AS Lower_Case_Name
77
78
        FROM EMP;
         -- find the date of 3 days later from hiredate.
79
         SELECT ENAME, HIREDATE, HIREDATE + INTERVAL '3' DAY
28 0
81
        AS Date_After_3_Days
82
         FROM EMP;
```

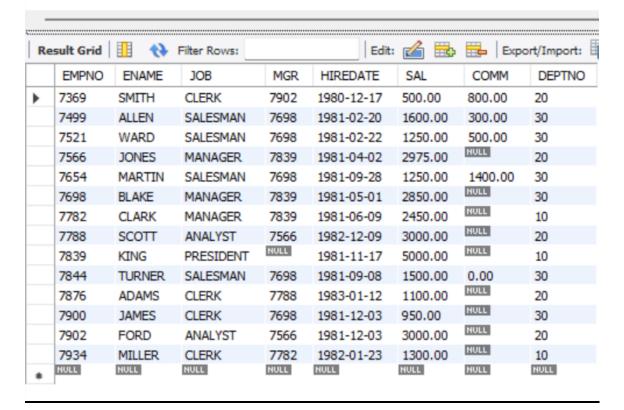
DBMS Lab

B.Tech. III Sem.



DBMS Lab B.Tech. III Sem.

1 • SELECT * FROM exp8.emp;



Conclusion

The SQL code successfully creates a relational database structure that allows for effective storage and retrieval of employee and department data. The queries implemented demonstrate the capability to analyze employee data, providing insights into salary distributions, employee counts, and unique hire dates. This database can serve as a foundation for more complex analyses and reporting, contributing to better management and decision-making in an organizational context.

Future enhancements could include more advanced data integrity checks, additional tables for roles or job histories, and more complex queries for deeper insights.