University of Petroleum

&

Energy Studies

SCHOOL OF COMPUTER SCIENCE

Name: Akshat Agarwal

Course: BTech CSE

SAP ID: 500118953

BATCH: 1

PRESENTED TO: Dr. Syed Sajid Hussain

Semester: 3

Experiment 6: Use of Inbuilt functions and relational algebra operation

Objective: To understand the use of inbuilt function and relational algebra with sql query.

1. **Create the following table.**
   - **A. EMP TABLE**

```
mysql> CREATE DATABASE LAB6;
Query OK, 1 row affected (0.03 sec)

mysql> USE LAB6;
Database changed
mysql> CREATE TABLE EMP (
    ->   EMPNO INT NOT NULL,
    ->   ENAME VARCHAR(20),
    ->   JOB VARCHAR(20),
    ->   MGR INT,
    ->   HIREDATE DATE,
    ->   SAL DECIMAL(10, 2),
    ->   COMM DECIMAL(10, 2),
    ->   DEPTNO INT,
    ->   PRIMARY KEY (EMPNO)
    -> );
Query OK, 0 rows affected (0.09 sec)
```

   - **B. DEPT TABLE**

```
mysql> CREATE TABLE DEPT (
    ->   DEPTNO INT NOT NULL,
    ->   DNAME VARCHAR(20),
    ->   LOC VARCHAR(20),
    ->   PRIMARY KEY (DEPTNO)
    -> );
Query OK, 0 rows affected (0.18 sec)
```

2. INSERTING DATA

A. EMP DATA

```
mysql> INSERT INTO EMP (EMPNO, ENAME, JOB, MGR, HIREDATE, SAL, COMM, DEPTNO) VALUES
    -> (7369, 'SMITH', 'CLERK', 7902, '1980-12-17', 800, NULL, 20),
    -> (7499, 'ALLEN', 'SALESMAN', 7698, '1981-02-20', 1600, 300, 30),
    -> (7521, 'WARD', 'SALESMAN', 7698, '1981-02-22', 1250, 500, 30),
    -> (7566, 'JONES', 'MANAGER', 7839, '1981-04-02', 2975, NULL, 20),
    -> (7654, 'MARTIN', 'SALESMAN', 7698, '1981-09-28', 1250, 1400, 30),
    -> (7698, 'BLAKE', 'MANAGER', 7839, '1981-05-01', 2850, NULL, 30),
    -> (7782, 'CLARK', 'MANAGER', 7839, '1981-06-09', 2450, NULL, 10),
    -> (7788, 'SCOTT', 'ANALYST', 7566, '1982-12-09', 3000, NULL, 20),
    -> (7839, 'KING', 'PRESIDENT', NULL, '1981-11-17', 5000, NULL, 10),
    -> (7844, 'TURNER', 'SALESMAN', 7698, '1981-09-08', 1500, 0, 30),
    -> (7876, 'ADAMS', 'CLERK', 7788, '1983-01-12', 1100, NULL, 20),
    -> (7900, 'JAMES', 'CLERK', 7698, '1981-12-03', 950, NULL, 30),
    -> (7902, 'FORD', 'ANALYST', 7566, '1981-12-03', 3000, NULL, 20),
    -> (7934, 'MILLER', 'CLERK', 7782, '1982-01-23', 1300, NULL, 10);
Query OK, 14 rows affected (0.14 sec)
Records: 14  Duplicates: 0  Warnings: 0
```

B. DEPT DATA

```
mysql> INSERT INTO DEPT (DEPTNO, DNAME, LOC) VALUES
    -> (10, 'ACCOUNTING', 'NEW YORK'),
    -> (20, 'RESEARCH', 'DALLAS'),
    -> (30, 'SALES', 'CHICAGO'),
    -> (40, 'OPERATIONS', 'BOSTON');
Query OK, 4 rows affected (0.14 sec)
Records: 4  Duplicates: 0  Warnings: 0
```

3. QUERIES

A. Retrieve average salary of all employees.

```
mysql> SELECT AVG(SAL) FROM EMP;
+-------------+
| AVG(SAL)    |
+-------------+
| 2073.214286 |
+-------------+
1 row in set (0.00 sec)
```

B. Retrieve the number of employees.

```
mysql> SELECT COUNT(*) FROM EMP;
+----------+
| COUNT(*) |
+----------+
|       14 |
+----------+
1 row in set (0.00 sec)
```

C. Retrieve distinct number of employees.

```
mysql> SELECT COUNT(DISTINCT ENAME) FROM EMP;
+------------------------+
| COUNT(DISTINCT ENAME) |
+------------------------+
|                     14 |
+------------------------+
1 row in set (0.12 sec)
```

D. Retrieve total salary of employee group by job.

```
mysql> SELECT JOB, SUM(SAL) FROM EMP GROUP BY JOB;
+-----------+----------+
| JOB       | SUM(SAL) |
+-----------+----------+
| CLERK     |  4150.00 |
| SALESMAN  |  5600.00 |
| MANAGER   |  8275.00 |
| ANALYST   |  6000.00 |
| PRESIDENT |  5000.00 |
+-----------+----------+
5 rows in set (0.00 sec)
```

E. Display the employee information with maximum salary.

```
mysql> SELECT * FROM EMP WHERE SAL = (SELECT MAX(SAL) FROM EMP);
+-------+-------+-----------+------+------------+---------+------+--------+
| EMPNO | ENAME | JOB       | MGR  | HIREDATE   | SAL     | COMM | DEPTNO |
+-------+-------+-----------+------+------------+---------+------+--------+
|  7839 | KING  | PRESIDENT | NULL | 1981-11-17 | 5000.00 | NULL |     10 |
+-------+-------+-----------+------+------------+---------+------+--------+
1 row in set (0.13 sec)
```

F. Find the highest paid employee in department 10.

```
mysql> SELECT * FROM EMP WHERE SAL = (SELECT MAX(SAL) FROM EMP WHERE DEPTNO = 10);
+-------+-------+-----------+------+------------+---------+------+--------+
| EMPNO | ENAME | JOB       | MGR  | HIREDATE   | SAL     | COMM | DEPTNO |
+-------+-------+-----------+------+------------+---------+------+--------+
|  7839 | KING  | PRESIDENT | NULL | 1981-11-17 | 5000.00 | NULL |     10 |
+-------+-------+-----------+------+------------+---------+------+--------+
1 row in set (0.00 sec)
```

G. Display the employee information with 2nd maximum salary.

```
mysql> SELECT * FROM EMP WHERE SAL = (SELECT MAX(SAL) FROM EMP WHERE SAL < (SELECT MAX(SAL) FROM EMP));
+-------+-------+---------+------+------------+---------+------+--------+
| EMPNO | ENAME | JOB     | MGR  | HIREDATE   | SAL     | COMM | DEPTNO |
+-------+-------+---------+------+------------+---------+------+--------+
|  7788 | SCOTT | ANALYST | 7566 | 1982-12-09 | 3000.00 | NULL |     20 |
|  7902 | FORD  | ANALYST | 7566 | 1981-12-03 | 3000.00 | NULL |     20 |
+-------+-------+---------+------+------------+---------+------+--------+
2 rows in set (0.00 sec)
```

H. List the emps whose sal is equal to the average of max and minimum.

```
mysql> SELECT * FROM EMP WHERE SAL = (SELECT (MAX(SAL)+MIN(SAL))/2 FROM EMP);
Empty set (0.12 sec)
```

I. List the emps who joined in the company on the same date with different EMPNO.

```
mysql> SELECT * FROM EMP E WHERE HIREDATE IN (SELECT HIREDATE FROM EMP WHERE EMPNO <> E.EMPNO);
+-------+-------+---------+------+------------+---------+------+--------+
| EMPNO | ENAME | JOB     | MGR  | HIREDATE   | SAL     | COMM | DEPTNO |
+-------+-------+---------+------+------------+---------+------+--------+
|  7900 | JAMES | CLERK   | 7698 | 1981-12-03 |  950.00 | NULL |     30 |
|  7902 | FORD  | ANALYST | 7566 | 1981-12-03 | 3000.00 | NULL |     20 |
+-------+-------+---------+------+------------+---------+------+--------+
2 rows in set (0.01 sec)
```

J. Display the employee names in upper and lower case.

```
mysql> SELECT UPPER(ENAME), LOWER(ENAME) FROM EMP;
+--------------+--------------+
| UPPER(ENAME) | LOWER(ENAME) |
+--------------+--------------+
| SMITH        | smith        |
| ALLEN        | allen        |
| WARD         | ward         |
| JONES        | jones        |
| MARTIN       | martin       |
| BLAKE        | blake        |
| CLARK        | clark        |
| SCOTT        | scott        |
| KING         | king         |
| TURNER       | turner       |
| ADAMS        | adams        |
| JAMES        | james        |
| FORD         | ford         |
| MILLER       | miller       |
+--------------+--------------+
14 rows in set (0.12 sec)
```

K. Find the date of 3 days later from hiredate.

```
mysql> SELECT HIREDATE, DATE_ADD(HIREDATE, INTERVAL 3 DAY) AS 3_DAYS_LATER FROM EMP;
+------------+--------------+
| HIREDATE   | 3_DAYS_LATER |
+------------+--------------+
| 1980-12-17 | 1980-12-20   |
| 1981-02-20 | 1981-02-23   |
| 1981-02-22 | 1981-02-25   |
| 1981-04-02 | 1981-04-05   |
| 1981-09-28 | 1981-10-01   |
| 1981-05-01 | 1981-05-04   |
| 1981-06-09 | 1981-06-12   |
| 1982-12-09 | 1982-12-12   |
| 1981-11-17 | 1981-11-20   |
| 1981-09-08 | 1981-09-11   |
| 1983-01-12 | 1983-01-15   |
| 1981-12-03 | 1981-12-06   |
| 1981-12-03 | 1981-12-06   |
| 1982-01-23 | 1982-01-26   |
+------------+--------------+
14 rows in set (0.00 sec)
```