

EXPERIMENT -6

Title: Use of Inbuilt functions and relational algebra operation

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

```
mysql> CREATE DATABASE EXP6;  
Query OK, 1 row affected (0.01 sec)
```

```
mysql> CREATE TABLE EMP (  
-> EMPNO INT PRIMARY KEY,  
-> ENAME VARCHAR(50),  
-> JOB VARCHAR(50),  
-> MGR INT,  
-> HIREDATE DATE,  
-> SAL DECIMAL(10, 2),  
-> COMM DECIMAL(10, 2),  
-> DEPTNO INT  
-> );  
Query OK, 0 rows affected (0.03 sec)
```

```

mysql> INSERT INTO EMP (EMPNO, ENAME, JOB, MGR, HIREDATE, SAL, COMM, DEPTNO)
-> VALUES
-> (7369, 'SMITH', 'CLERK', 7902, TO_DATE('17-DEC-1980', 'DD-MON-YYYY'), 800, NULL, 20),
-> (7499, 'ALLEN', 'SALESMAN', 7698, TO_DATE('20-FEB-1981', 'DD-MON-YYYY'), 1600, 300, 30),
-> (7521, 'WARD', 'SALESMAN', 7698, TO_DATE('22-FEB-1981', 'DD-MON-YYYY'), 1250, 500, 30),
-> (7566, 'JONES', 'MANAGER', 7839, TO_DATE('02-APR-1981', 'DD-MON-YYYY'), 2975, NULL, 20),
-> (7654, 'MARTIN', 'SALESMAN', 7698, TO_DATE('28-SEP-1981', 'DD-MON-YYYY'), 1250, 1400, 30),
-> (7698, 'BLAKE', 'MANAGER', 7839, TO_DATE('01-MAY-1981', 'DD-MON-YYYY'), 2850, NULL, 30),
-> (7782, 'CLARK', 'MANAGER', 7839, TO_DATE('09-JUN-1981', 'DD-MON-YYYY'), 2450, NULL, 10),
-> (7788, 'SCOTT', 'ANALYST', 7566, TO_DATE('09-DEC-1982', 'DD-MON-YYYY'), 3000, NULL, 20),
-> (7839, 'KING', 'PRESIDENT', NULL, TO_DATE('17-NOV-1981', 'DD-MON-YYYY'), 5000, NULL, 10),
-> (7844, 'TURNER', 'SALESMAN', 7698, TO_DATE('08-SEP-1981', 'DD-MON-YYYY'), 1500, 0, 30),
-> (7876, 'ADAMS', 'CLERK', 7788, TO_DATE('12-JAN-1983', 'DD-MON-YYYY'), 1100, NULL, 20),
-> (7900, 'JAMES', 'CLERK', 7698, TO_DATE('03-DEC-1981', 'DD-MON-YYYY'), 950, NULL, 30),
-> (7902, 'FORD', 'ANALYST', 7566, TO_DATE('03-DEC-1981', 'DD-MON-YYYY'), 3000, NULL, 20),
-> (7934, 'MILLER', 'CLERK', 7782, TO_DATE('23-JAN-1982', 'DD-MON-YYYY'), 1300, NULL, 10);
ERROR 1305 (42000): FUNCTION exp6.TO_DATE does not exist
mysql> DROP TABLE EMP;
Query OK, 0 rows affected (0.04 sec)

mysql> CREATE TABLE EMP (
-> EMPNO INT PRIMARY KEY,
-> ENAME VARCHAR(50),
-> JOB VARCHAR(50),
-> MGR INT,
-> HIREDATE VARCHAR(10),
-> SAL DECIMAL(10, 2),
-> COMM DECIMAL(10, 2),
-> DEPTNO INT
-> );
Query OK, 0 rows affected (0.03 sec)

```

```

mysql> CREATE TABLE EMP (
-> EMPNO INT PRIMARY KEY,
-> ENAME VARCHAR(50),
-> JOB VARCHAR(50),
-> MGR INT,
-> HIREDATE VARCHAR(10),
-> SAL DECIMAL(10, 2),
-> COMM DECIMAL(10, 2),
-> DEPTNO INT
-> );
Query OK, 0 rows affected (0.03 sec)

mysql> INSERT INTO EMP (EMPNO, ENAME, JOB, MGR, HIREDATE, SAL, COMM, DEPTNO)
-> VALUES
-> (7369, 'SMITH', 'CLERK', 7902, '17-DEC-80', 800, NULL, 20),
-> (7499, 'ALLEN', 'SALESMAN', 7698, '20-FEB-81', 1600, 300, 30),
-> (7521, 'WARD', 'SALESMAN', 7698, '22-FEB-81', 1250, 500, 30),
-> (7566, 'JONES', 'MANAGER', 7839, '02-APR-81', 2975, NULL, 20),
-> (7654, 'MARTIN', 'SALESMAN', 7698, '28-SEP-81', 1250, 1400, 30),
-> (7698, 'BLAKE', 'MANAGER', 7839, '01-MAY-81', 2850, NULL, 30),
-> (7782, 'CLARK', 'MANAGER', 7839, '09-JUN-81', 2450, NULL, 10),
-> (7788, 'SCOTT', 'ANALYST', 7566, '09-DEC-82', 3000, NULL, 20),
-> (7839, 'KING', 'PRESIDENT', NULL, '17-NOV-81', 5000, NULL, 10),
-> (7844, 'TURNER', 'SALESMAN', 7698, '08-SEP-81', 1500, 0, 30),
-> (7876, 'ADAMS', 'CLERK', 7788, '12-JAN-83', 1100, NULL, 20),
-> (7900, 'JAMES', 'CLERK', 7698, '03-DEC-81', 950, NULL, 30),
-> (7902, 'FORD', 'ANALYST', 7566, '03-DEC-81', 3000, NULL, 20),
-> (7934, 'MILLER', 'CLERK', 7782, '23-JAN-82', 1300, NULL, 10);
Query OK, 14 rows affected (0.01 sec)
Records: 14 Duplicates: 0 Warnings: 0

```

```
mysql> CREATE TABLE DEPT (
->     DEPTNO INT PRIMARY KEY,
->     DNAME VARCHAR(50),
->     LOC VARCHAR(50)
-> );
Query OK, 0 rows affected (0.02 sec)

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 (0.01 sec)
Records: 4  Duplicates: 0  Warnings: 0
```

```
mysql> SELECT * FROM EMP;
+-----+-----+-----+-----+-----+-----+-----+-----+
| EMPNO | ENAME  | JOB      | MGR  | HIREDATE | SAL      | COMM      | DEPTNO |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 7369 | SMITH  | CLERK    | 7902 | 17-DEC-80 | 800.00   | NULL      | 20      |
| 7499 | ALLEN  | SALESMAN | 7698 | 20-FEB-81 | 1600.00  | 300.00    | 30      |
| 7521 | WARD   | SALESMAN | 7698 | 22-FEB-81 | 1250.00  | 500.00    | 30      |
| 7566 | JONES  | MANAGER  | 7839 | 02-APR-81 | 2975.00  | NULL      | 20      |
| 7654 | MARTIN | SALESMAN | 7698 | 28-SEP-81 | 1250.00  | 1400.00   | 30      |
| 7698 | BLAKE  | MANAGER  | 7839 | 01-MAY-81 | 2850.00  | NULL      | 30      |
| 7782 | CLARK  | MANAGER  | 7839 | 09-JUN-81 | 2450.00  | NULL      | 10      |
| 7788 | SCOTT  | ANALYST  | 7566 | 09-DEC-82 | 3000.00  | NULL      | 20      |
| 7839 | KING   | PRESIDENT | NULL | 17-NOV-81 | 5000.00  | NULL      | 10      |
| 7844 | TURNER | SALESMAN | 7698 | 08-SEP-81 | 1500.00  | 0.00      | 30      |
| 7876 | ADAMS  | CLERK    | 7788 | 12-JAN-83 | 1100.00  | NULL      | 20      |
| 7900 | JAMES  | CLERK    | 7698 | 03-DEC-81 | 950.00   | NULL      | 30      |
| 7902 | FORD   | ANALYST  | 7566 | 03-DEC-81 | 3000.00  | NULL      | 20      |
| 7934 | MILLER | CLERK    | 7782 | 23-JAN-82 | 1300.00  | NULL      | 10      |
+-----+-----+-----+-----+-----+-----+-----+-----+
14 rows in set (0.01 sec)
```

1.Retrieve average salary of all employees.

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

2. Retrieve the number of employees.

```
mysql> SELECT COUNT(*) FROM EMP;
+-----+
| COUNT(*) |
+-----+
| 14 |
+-----+
1 row in set (0.01 sec)
```

3. Retrieve distinct number of employees.

```
mysql> SELECT COUNT(DISTINCT ENAME) FROM EMP;
+-----+
| COUNT(DISTINCT ENAME) |
+-----+
|          14          |
+-----+
1 row in set (0.01 sec)
```

4. 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)
```

5. Display the employee information with 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 | 09-DEC-82 | 3000.00 | NULL | 20     |
| 7902 | FORD  | ANALYST  | 7566 | 03-DEC-81 | 3000.00 | NULL | 20     |
+-----+-----+-----+-----+-----+-----+-----+-----+
2 rows in set (0.00 sec)
```

6. Find the highest paid employee in department 10.

```
mysql> SELECT * FROM EMP WHERE DEPTNO = 10 AND SAL = (SELECT MAX(SAL) FROM EMP WHERE DEPTNO = 10);
+-----+-----+-----+-----+-----+-----+-----+-----+
| EMPNO | ENAME | JOB      | MGR | HIREDATE | SAL      | COMM | DEPTNO |
+-----+-----+-----+-----+-----+-----+-----+-----+
| 7839 | KING  | PRESIDENT | NULL | 17-NOV-81 | 5000.00 | NULL | 10     |
+-----+-----+-----+-----+-----+-----+-----+-----+
1 row in set (0.00 sec)
```

7. 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 (0.00 sec)
```

8. List the emps who joined in the company on the same date.

```
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 | 03-DEC-81 | 950.00  | NULL | 30     |
| 7902 | FORD  | ANALYST  | 7566 | 03-DEC-81 | 3000.00 | NULL | 20     |
+-----+-----+-----+-----+-----+-----+-----+-----+
2 rows in set (0.00 sec)
```

9. 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 (0.00 sec)
```

10. find the date of 3 days later from hiredate.

```
mysql> SELECT HIREDATE, (HIREDATE + 3) FROM EMP;
+-----+-----+
| HIREDATE | (HIREDATE + 3) |
+-----+-----+
| 17-DEC-80 | 20              |
| 20-FEB-81 | 23              |
| 22-FEB-81 | 25              |
| 02-APR-81 | 5               |
| 28-SEP-81 | 31              |
| 01-MAY-81 | 4               |
| 09-JUN-81 | 12              |
| 09-DEC-82 | 12              |
| 17-NOV-81 | 20              |
| 08-SEP-81 | 11              |
| 12-JAN-83 | 15              |
| 03-DEC-81 | 6               |
| 03-DEC-81 | 6               |
| 23-JAN-82 | 26              |
+-----+-----+
14 rows in set, 14 warnings (0.01 sec)
```

NAME - NIPUN AGRAWAL

BATCH – 2

SAP ID - 500119472

