7-8-25

TASK-3: Developing Queries with DML Single-row functions & Operat

Aim:- To perform the Query processing on databases for different retrieval results of queries using DML, DRL Single-row operations using aggregate, data, string, inden functions, set clauses and operations

procedure:-

Create table for employee Schema and insert around retail_store_retail_store_employees data in this relatio perform multi row functions.

Aggregative operators: In addition to simply retrieving data we often want to perform some computation or summa rization. SQL allows the use of arithmetic expressions we now consider a powerful class of constructs for computing aggregate values such as MIN and SUM.

① Count: Count following by a column name returns the count of the tuple in that column. If DISTINCT keyword is use then it will return only the count of unique tuple in the column. otherwise, it will return count of all the tuples Count (*) indicates all the tuples of the column.

Syntax:- COUNT(column name)

EX:- SELECT COUNT(*) FROM retail_store_employees;

② SUM:- SUM followed by a column name returns the sum of all the values in that column.

Syntax:- SUM(column name)

Ex:- SELECT SUM (salary) FROM retail_store_employees;

③ AVG: AVG followed by a column name returns the average value of that column values.

Syntax:- AVG(n1, n2....)

④ MAX:- MAX followed by a column name returns the maximum value of that column.

Syntax:- MAX(column name)

⑤MIN: MIN followed by column name returns the minimum value of that column.

Syntax: MIN (column name)

SQL String Functions:

String functions are used to perform an operation on input string and return an output string. Following are the string functions defined in SQL:

①UPPER()

Query: SELECT UPPER (retailstore_employee_name) FROM retail_store_employees WHERE retailstore_employee_id = 1;

②LOWER()

Query: SELECT LOWER (retailstore_employee_name) FROM retail_store_employees WHERE retailstore_employee_id = 1;

③LENGTH()

Query: SELECT LENGTH (retailstore_employee_name, 1, 5) FROM retail_store_employees WHERE retailstore_employee_id = 1;

④SUBSTER()

Query: SELECT SUBSTER (retailstore_employee_name, 1, 5) FROM retail_store_employees WHERE retailstore_employee_id = 1;

⑤CONCAT()

Query: SELECT CONCAT (retailstore_employee_name, '', department) FROM retail_store_employees WHERE retailstore_employee_id = 1;

SQL Date and Time functions

The date & time functions are built in functions in the SQL. These functions can be used in SQL queries to perform various date and time operations, such as filtering records based on dates, calculating date differences and formatting dates for display purposes.

If storing a date or a date and time value in a database, My SQL offers the following data types

DATE	YYYY-MM-DD
DATETIME	YYYY-MM-DD/HH:MI:SS
TIMESTAMP	YYYY-MM-DD/HH:MI:SS
YEAR	YYYY or YY

CURDATE()

Query: SELECT CURRENTDATE from dual;

CURTIME()

Query: SELECT CURRENT_TIME() from dual;

ADDDATE(DATE,DAYS)

SQL > SELECT ADDDATE('2018-08-01',31);

ADDTIME(exp1,exp2)

SQL > SELECT ADDTIME('2018-08-01 23:59:59.9999','1:1:1.000002');

DAY OF MONTH (date)

SQL > SELECT DAY OF MONTH ('2018-02-15');

DAY OF WEEK (date)

SQL > SELECT DAY OF WEEK ('2018-02-15');

DAY of YEAR(date)

SQL > SELECT DAY OF YEAR ('2018-08-05');

MONTH (date)

SQL > SELECT MONTH ('2018-08-01');

TIME(expr)

SQL > SELECT TIME ('2018-08-01 11:33:25');

Sysdate:

SQL > SELECT SYSDATE FROM DUAL;

next_day:

SQL > SELECT NEXT_DAY(SYSDATE,'WED') FROM DUAL;

add_months:

SQL> SELECT ADD_MONTHS(SYSDATE, 8) FROM DUAL;

last_day:

SQL> SELECT LAST_DAY(SYSDATE) FROM DUAL;

Months_between:

SQL> SELECT MONTHS_BETWEEN(SYSDATE, HIREDATE) FROM EMP;

Least:

SQL> SELECT LEAST('10-JAN-07', '12-OCT-07') FROM DUAL;

Greatest:

SQL> SELECT GREATEST('10-JAN-07', '12-OCT-07') FROM DUAL;

Trunc:

SQL> SELECT TRUNC(SYSDATE, 'DAY') FROM DUAL;

Round:

SQL> SELECT ROUND(SYSDATE, 'DAY') FROM DUAL;

to_char:

SQL> Select to_char(sysdate, "dd/mm/yy") from dual;

to_date:

SQL> Select to date (sysdate, "dd/mm/yy") from dual;

CHARACTER FUNCTION:

initcap(char): select initcap("hello") from dual;

lower(char): select lower('HELLO') from dual;

upper(char): select upper('hello') from dual;

ltrim(char, [set]): select ltrim('cseit', 'cse') from dual;

rtrim(char, [set]): select rtrim('cseit', 'it') from dual;

replace(char, search): select replace('Jack and Jue', 'j', 'bl')
from dual;

STRING FUNCTIONS:

Concat: returns char1 concatenated with char2. Both char1 and char2 can be any of the datatypes.

SQL> SELECT CONCAT('ORACLE', 'CORPORATION') FROM DUAL;

ORACLECORPORATION

LPad: returns expr1, left-padded to length n characters with the sequence of characters in expr2

SQL> SELECT LPAD ('ORACLE', 15, '*') FROM DUAL;

RPad: returns expr1, right-padded to length n characters with expr2, replicated as many times as necessary

SQL> SELECT RPAD ('ORACLE', 15, '*') FROM DUAL;

Ltrim: returns a character expression after removing leading blanks

SQL> SELECT LTRIM ('SSMITHSS', 'S') FROM DUAL;

Rtrim: returns a character expression after truncating all trailing blanks

SQL> SELECT RTRIM ('SSMITHSS', 'S') FROM DUAL;

Lower: returns a character expression after converting uppercase character data to lowercase.

SQL> SELECT LOWER ('DBMS') FROM DUAL;

Upper: returns a character expression with lowercase character data converted converted to UPPER case.

SQL> SELECT UPPER ('dbms') FROM DUAL;

Length: returns the no of characters, rather than the no. of bytes, of the given string expression, excluding trailing blanks

SQL> SELECT LENGTH ('DATABASE') FROM DUAL;

Substr: returns part of a char, binary, text or image expression

SQL> SELECT SUBSTR ('retail store employee name', 1, 4) FROM retail-store.employees;

Instr: INSTR functions search string for substring. The function returns an integer indicating the position of the character in string that is the first character of this occurrence.

SQL> SELECT INSTR ('CORPORATE FLOOR', 'OR', 3, 2) FROM DUAL;

Single Row operators:

① IS NULL

Query: SELECT * FROM retail_store_employees WHERE Salaryary IS NULL;

② IS NOT NULL

Query: SELECT * FROM retail_store_employees WHERE salaryary IS NOT NULL;

③ LIKE

Query: SELECT * FROM retail_store_employees WHERE retailstore_employee_name LIKE '%Ihon%';

④ NOT LIKE

Query: SELECT * FROM retail_store_employees WHERE retailstore_employee_name NOT LIKE '%John%';

⑤ BETWEEN

Query: SELECT * FROM retail_store_employees WHERE Salaryary BETWEEN 50000 AND 100000;

Output:-

⇒ SQL > select * from retail_store_employees;

Retail_store_employee_ID STORE_ID
Retail_store_emp Salary department

STORE_phoneno EMPloyee_PHNO

30111	2001 anirudh		
9123456789	9876501111	50000	Cashier
30222	2001 bhavani	40000	supervisor
9123456789	9876502222		
30333	2001 Pradeep	60000	hr
9123456789	9876503333		
30444	2001 keerthana	35000	Salesman
9123456789	9876504444		

=> SQL > Select count (*) from retail_store_employees;
count (*)
OP: 4

=> SQL > Select sum (salary) from retail_store_employees;
sum(salary) OP: 185000

=> SQL > Select avg (salary) from retail_store_employees;
AVG(salary) OP: 46250

=> SQL > Select max(salary) from retail_store_employees;
max(salary) OP: 60000

=> SQL > select min (salary) from retail_store_employees;
MIN(salary) OP: 35000

=> SQL > select upper (retail_store_employee_name) from retail_store_employees where retail_store_employee_id = 3033
~~Lower~~ (retail_store) OP: ~~Pradeep~~ PRADEEP

=> SQL > select lower (retail_store_employee_name) from retail_store_employees where retail_store_employee_id = 30333
LOWER (retail_st) OP: Pradeep

=> SQL > select length (retail_store_employee_name) from retail_store_employees where retail_store_employee_id = 30333
LENGTH (Retail_store_employee_name) OP: 7

=> SQL > select Substr (retail_store_employee_name, 1, 4) from retail_store_employees where retail_store_employee_id = 30111;
SUBSTR (Retail_st) OP: anir

=> SQL > select concat ('retail_store_employee_name', 'salary') from retail_store_employees where retail_store_emp_id = 30444;
CONCAT (Retail_store_employee_name~~salary~~
OP: retail_store_employee_name salary

=> SQL > select Sysdate from dual;
SYSDATE OP: 27-AUG-25

=> SQL > Select next_day (sysdate, 'wed') from dual;
NEXT_DAY OP: 03-SEP-25

=> SQL > Select add_months (sysdate, 2) from dual;
ADD_MONTH OP: 27-OCT-25

⇒ SQL > select last_day(sysdate) from dual;
LAST_DAY OP: 31-AUG-25

⇒ SQL > Select least('10-jan-07', '12-oct-07') from dual;
LEAST('10) OP:- 10-Jan-07

⇒ SQL > select greatest('10-jan-07', '12-oct-07') from dual;
GREATEST() OP: 12-OCT-07

⇒ SQL > slect trunc(sysdate, 'day') from dual;
TRUNC(SYS OP: 24-AUG-25

⇒ SQL > select round(sysdate, 'day') from dual;
ROUND(SYS OP: 31-AUG-25

⇒ SQL > select to_char(sysdate, 'dd/mm/yy') from dual;
TO_CHAR(OP: 27/08/25

⇒ SQL > Select to_date(sysdate, 'dd/mm/yy') from dual;
TO_DATE(S OP: 27-AUG-25

⇒ SQL > select contact('oracle', 'corporation') from dual;
CONCAT('ORACLE', 'CORPORATION') OP: oracle corporation

⇒ SQL > select lpad('oracle', '15', *) from dual;
LPAD(oracle', '15', *') OP: ********oracle

⇒ SQL > select rpad('oracle', '15', *') from dual;
RPAD('ORACLE', 15', *') OP: oracle*******

⇒ SQL > select ltrim('ssmithss', 's') from dual;
LTRIM(OP: mithss

⇒ SQL > select lower('dbms') from dual;
LOWER OP: dbms

⇒ ~~SQL > select rtrim('ssmithss'~~

⇒ SQL > select upper('dbms') from dual;
UPPE OP: DBMS

⇒ SQL > select rtrim('ssmithss', 's') from dual;
RTRIM(OP: SSmith

⇒ SQL > select length('database') from dual;
LENGTH('DATABASE') OP: 8

⇒ SQL > select Substr('abcdefghij', 3, 4) from dual;
SUBS OP: cdef

TASK 3 OP

```
SQL> select * from retail_store_employees;

RETAIL_STORE_EMPLOYEE_ID  STORE_ID RETAIL_STORE_EMP    SALARY DEPARTMENT
------------------------ --------- --------------- ---------- ---------------
STORE_PHONENO EMPLOYEE_PHONENO
------------ ---------------
          30111      2001 anirudh          50000 cashier
 9123456789     9876501111

          30222      2001 bhavani          40000 supervisor
 9123456789     9876502222

          30333      2001 pradeep          60000 hr
 9123456789     9876503333

          30444      2001 keerthana        35000 salesman
 9123456789     9876504444
```

```
SQL> select count(*) from retail_store_employees;

 COUNT(*)
----------
         4
```

```
SQL> select sum(salary) from retail_store_employees;

SUM(SALARY)
-----------
    185000

SQL> select avg(salary) from retail_store_employees;

AVG(SALARY)
-----------
     46250

SQL> select max(salary) from retail_store_employees;

MAX(SALARY)
-----------
     60000

SQL> select min(salary) from retail_store_employees;

MIN(SALARY)
-----------
     35000
```

SQL> select upper(retail_store_employee_name) from retail_store_employees where retail_store_employee_id=30333;

UPPER(RETAIL_ST)

PRADEEP

SQL> select lower(retail_store_employee_name) from retail_store_employees where retail_store_employee_id=30333;

LOWER(RETAIL_ST)

---

pradeep

SQL> select length(retail_store_employee_name) from retail_store_employees where retail_store_employee_id=30333;

LENGTH(RETAIL_STORE_EMPLOYEE_NAME)

7

SQL> select substr(retail_store_employee_name,1,4) from retail_store_employees where retail_store_employee_id=30111;

SUBSTR(RETAIL_ST)

---

anir

SQL> select concat('retail_store_employee_name','salary') from retail_store_employees where retail_store_employee_id=30444;

CONCAT('RETAIL_STORE_EMPLOYEE_NA

---

retail_store_employee_namesalary

SQL> select sysdate from dual;

SYSDATE

27-AUG-25

SQL> select next_day(sysdate,'wed') from dual;

NEXT_DAY(

03-SEP-25

SQL> select add_months(sysdate,2) from dual;

ADD_MONTH

27-OCT-25

SQL> select last_day(sysdate) from dual;

LAST_DAY(

31-AUG-25

SQL> select least('10-jan-07','12-oct-07') from dual;

LEAST('10

10-jan-07

SQL> select greatest('10-jan-07','12-oct-07') from dual;

GREATEST(

12-oct-07

SQL> select trunc(sysdate,'day') from dual;

TRUNC(SYS

24-AUG-25

```
SQL> select round(sysdate,'day') from dual;

ROUND(SYS
---------
31-AUG-25

SQL> select to_char(sysdate,'dd\mm\yy') from dual;

TO_CHAR(
--------
27\08\25

SQL> select to_date(sysdate,'dd\mm\yy') from dual;

TO_DATE(S
---------
27-AUG-25

SQL> select concat('oracle','corpuration') from dual;

CONCAT('ORACLE','CORPURATION')
------------------------------
oraclecorpuration
```

```
SQL> select lpad('oracle',15,'*') from dual;

LPAD('ORACLE',15,'*')
---------------
*********oracle

SQL> select rpad('oracle',15,'*') from dual;

RPAD('ORACLE',15,'*')
---------------
oracle*********

SQL> select ltrim('ssmithss','s') from dual;

LTRIM(
------
mithss

SQL> select lower('dbms') from dual;

LOWE
----
dbms
```

```
SQL> select upper('dbms') from dual;

UPPE
----
DBMS

SQL> select rtrim('ssmithss','s') from dual;

RTRIM(
------
ssmith

SQL> select length('database') from dual;

LENGTH('DATABASE')
------------------
                 8

SQL> select substr('abcdefghij',3,4) from dual;

SUBS
----
cdef

SQL> select instr('corporate floor','or',3,2) from dual;
```

INSTR('CORPORATEFLOOR','OR',3,2)

14

SQL> select * from retail_store_employees where salary is null;

no rows selected

SQL> select * from retail_store_employees where salary is not null;

RETAIL_STORE_EMPLOYEE_ID STORE_ID RETAIL_STORE_EMP SALARY DEPARTMENT

STORE_PHONENO EMPLOYEE_PHONENO

30111 2001 anirudh 50000 cashier

9123456789 9876501111

30222 2001 bhavani 40000 supervisor

9123456789 9876502222

30333 2001 pradeep 60000 hr

9123456789 9876503333

30444 2001 keerthana 35000 salesman

9123456789 9876504444

SQL> select * from retail_store_employees where retail_store_employee_name like '%anirudh

RETAIL_STORE_EMPLOYEE_ID STORE_ID RETAIL_STORE_EMP SALARY DEPARTMENT

STORE_PHONENO EMPLOYEE_PHONENO

30111 2001 anirudh 50000 cashier

9123456789 9876501111

SQL> select * from retail_store_employees where retail_store_employee_name not like '%anirudh%

RETAIL_STORE_EMPLOYEE_ID STORE_ID RETAIL_STORE_EMP SALARY DEPARTMENT

STORE_PHONENO EMPLOYEE_PHONENO

30222 2001 bhavani 40000 supervisor

9123456789 9876502222

30333 2001 pradeep 60000 hr

9123456789 9876503333

30444 2001 keerthana 35000 salesman

9123456789 9876504444

SQL> select * from retail_store_employees where salary between 30000 and 70000;

RETAIL_STORE_EMPLOYEE_ID STORE_ID RETAIL_STORE_EMP SALARY DEPARTMENT

STORE_PHONENO EMPLOYEE_PHONENO

30111 2001 anirudh 50000 cashier

9123456789 9876501111

30222 2001 bhavani 40000 supervisor

9123456789 9876502222

30333 2001 pradeep 60000 hr

9123456789 9876503333

30444 2001 keerthana 35000 salesman

9123456789 9876504444

⇒ SQL > Select instr ('corporate floor', 'or', 3, 2) from dual;

INSTR('CORPORATEFLOOR','OR',3,2) OP: 14

⇒ SQL > select * from retail_store_employees where salary = null; OP: no rows selected.

⇒ SQL > Select * from retail store employees where salary is not null;

OP: Total table

⇒ SQL > select * from retail_Store_employees where retail_Store_employee_name like '%anisudh%';

Retail_st EMP_ID STORE_ID Rt_St_emp salary department

STORE_PHNO EMP_PHNO

OP: 30111 2001 anisudh 50000 cashier

9123456789 9876501111

⇒ SQL > Select * from st_st_emp where st_st_emp_name not like '%anisudh%', '%pradeep%', 'keerthana%';

OP: 30222 2001 bhavani 40000 Supervisor

9123456789 9876502222

⇒ Select * from st_st_emp where salary between 60000 to 70000;

RTST_EMP_ID STR_ID RT_ST_EMP SALARY DEPARTMENT

STORE_PHONENO EMPLOYEE_PHONENO

30333 2001 Pradeep 60000 hr

9123456789 9876503333

Result: Thus the Developing of queries with DML Single-row functions and operations has been executed successfully.

VEL TECH	
EX NO.	3
PERFORMANCE (5)	5
RESULT AND ANALYSIS (5)	5
VIVA VOCE (5)	5
RECORD (5)	5 5+5=20
TOTAL (20)	14
SIGN WITH DATE	14/8