Experiment No. : 4

Title: Implementation of different types of functions with suitable examples:
- □ Number Function   □ Aggregate Function
- □ Character Function   □ Conversion Function
- □ Date Function

Objectives:

To understand and implement various types of function in SQL..

Key Concepts: min, max, sum, avg, count

Theory:

NUMBER FUNCTION:

Abs(n) :   Select abs(-15) from dual;

Exp(n):   Select exp(4) from dual;

Power(m,n):   Select power(4,2) from dual;

Mod(m,n):   Select mod(10,3) from dual;

Round(m,n):   Select round(100.256,2) from dual;

Trunc(m,n):   Select trunc(100.256,2) from dual;

Sqrt(m,n);   Select sqrt(16) from dual

AGGREGATE FUNCTION:

An aggregate function summarizes the results of an expression over a number of rows, returning a single value. The general syntax for most of the aggregate functions is as follows:

aggregate_function([DISTINCT | ALL] expression)

The syntax elements are:

aggregate_function: Gives the name of the function. e.g., SUM, COUNT, AVG, MAX, MIN

DISTINCT:LSpecifies that the aggregate function should consider only distinct values of the argument expression.

ALL: Specifies that the aggregate function should consider all values, including all duplicate values, of the argument expression. The default is ALL.

expression: Specifies a column, or any other expression, on which you want to perform the aggregation.

1. **MIN( ) :** MIN followed by column name returns the minimum value of that column.
Syntax: MIN (Column name)
Example: SELECT MIN (Sal) FROM emp;

2. **MAX( ) :** MAX followed by a column name returns the maximum value of that column.
Syntax: MAX (Column name)
Example: SELECT MAX (Sal) FROM emp;

3. **AVG( ) :** AVG followed by a column name returns the average value of that column values.
Syntax: AVG (Column name)

Example: SELECT AVG (Sal) FROM emp;

4. **SUM( ) :** SUM followed by a column name returns the sum of all the values in that column.
Syntax: SUM (Column name)
Example: SELECT SUM (Sal) FROM emp;

5. **COUNT( ) :** COUNT following by a column name returns the count of tuple in that column.
   count (*) indicates all the tuples of the column.
Syntax: COUNT (Column name)
Example: SELECT COUNT (Sal) FROM emp;

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;

AGGREGATE FUNCTION:

To_char: TO_CHAR (number) converts n to a value of VARCHAR2 data type, using the optional number format fmt. The value n can be of type NUMBER, BINARY_FLOAT, or BINARY_DOUBLE.
Example:SQL>select to_char(65,'RN')from dual;   ANS: LXV

To_number: TO_NUMBER converts expr to a value of NUMBER data type.
Example: SQL> Select to_number ('1234.64') from Dual;   ANS: 1234.64

To_date:TO_DATE converts char of CHAR, VARCHAR2, NCHAR, or NVARCHAR2 data type to a value of DATE data type.
Example: SQL>SELECT TO_DATE('January 15, 1989, 11:00 A.M.')FROM DUAL;
ANS-   TO_DATE

              15-JAN-89

STRING FUNCTIONS:

concat: CONCAT returns char1 concatenated with char2. Both char1 and char2 can be any of the datatypes
SQL>SELECT CONCAT('ORACLE','CORPORATION')FROM DUAL;
      ANS-ORACLECORPORATION

Lpad: LPAD returns expr1, left-padded to length n characters with the sequence of characters in expr2.
Example: SQL>SELECT LPAD('ORACLE',15,'*')FROM DUAL;
      ANS- *********ORACLE

Rpad: RPAD returns expr1, right-padded to length n characters with expr2, replicated as many times as necessary.
Example: SQL>SELECT RPAD ('ORACLE',15,'*')FROM DUAL;
        ANS-ORACLE*********

Ltrim: Returns a character expression after removing leading blanks.
Example: SQL>SELECT LTRIM('SSMITHSS','S')FROM DUAL;   ANS-MITHSS

Rtrim: Returns a character string after truncating all trailing blanks
Example: SQL>SELECT RTRIM('SSMITHSS','S')FROM DUAL;   ANS-SSMITH

Lower: Returns a character expression after converting uppercase character data to lowercase.
Example: SQL>SELECT LOWER('DBMS')FROM DUAL;   ANS-dbms

Upper: Returns a character expression with lowercase character data converted to uppercase
Example: SQL>SELECT UPPER('dbms')FROM DUAL;   ANS-DBMS

Length: Returns the number of characters, rather than the number of bytes, of the given string expression, excluding trailing blanks.
Example: SQL>SELECT LENGTH('DATABASE')FROM DUAL;   ANS-8

Substr: Returns part of a character, binary, text, or image expression.
Example: SQL>SELECT SUBSTR('ABCDEFGHIJ'3,4)FROM DUAL;   ANS-CDEF

Instr: The 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.
Example:SQL>SELECT INSTR('CORPORATE FLOOR','OR',3,2)FROM DUAL;   ANS-14

DATE FUNCTIONS:
Sysdate: SQL>SELECT SYSDATE FROM DUAL;   ANS-29-DEC-08
next_day: SQL>SELECT NEXT_DAY(SYSDATE,'WED')FROM DUAL;   ANS-05-JAN-09
add_months: SQL>SELECT ADD_MONTHS(SYSDATE,2)FROM DUAL;   ANS-28-FEB-09
last_day: SQL>SELECT LAST_DAY(SYSDATE)FROM DUAL;   ANS-31-DEC-08
months_between: SQL>SELECT MONTHS_BETWEEN(SYSDATE,HIREDATE)FROM EMP;   ANS-4
Least: SQL>SELECT LEAST('10-JAN-07','12-OCT-07')FROM DUAL;   ANS-10-JAN-07
Greatest: SQL>SELECT GREATEST('10-JAN-07','12-OCT-07')FROM DUAL;
                        ANS-10-JAN-07
Trunc: SQL>SELECT TRUNC(SYSDATE,'DAY')FROM DUAL;   ANS-28-DEC-08
Round: SQL>SELECT ROUND(SYSDATE,'DAY')FROM DUAL;   ANS-28-DEC-08
to_char: SQL> select to_char(sysdate, "dd\mm\yy") from dual;   ANS-24-mar-05.
to_date: SQL> select to date (sysdate, "dd\mm\yy") from dual;   ANS-24-mar-05.

LAB WORK:

Q1. Perform following queries on bank database:
1. Find the average account balance at the Perryridge branch.
2. Find maximum amount of any loan in the bank
3. List Numbers of accounts with balances between 700 and 900
4. List total number of account holders in the 'Capital Bank' Branch
5. List total number of unique Customer city names
6. Find the number of branches that currently have loans.

Q2. Create a table EMPLOYEE with following schema: (Emp_no, E_name, E_address, E_ph_no, Dept_no, Dept_name,Job_id, Designation , Salary, joiningdate)
Write SQL statements for the following query.
1. List the E_no, E_name, Salary of all employees working for MANAGER.
2. Display all the details of the employee whose salary is more than the Sal of any IT PROFF..
3. List the employees in the ascending order of Designations of those joined after 1981.
4. List the employees along with their Experience and Daily Salary.
5. List the employees who are either 'CLERK' or 'ANALYST' .
6. List the employees who joined on 1-MAY-81, 3-DEC-81, 17-DEC-81,19-JAN-80 .
7. List the employees who are working for the Deptno 10 or20.
8. List the Enames those are starting with 'S' .
9. Dislay the name as well as the first five characters of name(s) starting with 'H'
10. List all the emps except 'PRESIDENT' & 'MGR" in asc order of Salaries.