

## EXP 11

# FUNCTIONS

**AIM:** To execute FUNCTIONS in sql

### 1) CREATING FUNCTION AND CALLING IT:

```
SQL> create or replace function adder(n1 in number, n2 in number)
  2  return number
  3  is
  4  n3 number(8);
  5  begin
  6  n3 := n1+n2;
  7  return n3;
  8  end;
  9  /
```

```
SQL> DECLARE
  2  n3 number(2);
  3  BEGIN
  4  n3 := adder(11,22);
  5  dbms_output.put_line('Addition is: ' || n3);
  6  END;
  7  /
```

PL/SQL procedure successfully completed.

```
SQL> DECLARE
  2  a number;
  3  b number;
  4  c number;
  5  FUNCTION findMax(x IN number, y IN number)
  6  RETURN number
  7  IS
  8  z number;
  9  BEGIN
 10  IF x > y THEN
 11  z:= x;
 12  ELSE
 13  z:= y;
 14  END IF;
 15
 16  RETURN z;
 17  END;
 18  BEGIN
 19  a:= 23;
 20  b:= 45;
 21
 22  c := findMax(a, b);
 23  dbms_output.put_line(' Maximum of (23,45): ' || c);
 24  END;
 25  /
```

## 2)PL/SQL RECURSIVE FUNCTION

```
SQL> DECLARE
  2     num number;
  3     factorial number;
  4
  5 FUNCTION fact(x number)
  6 RETURN number
  7 IS
  8     f number;
  9 BEGIN
 10     IF x=0 THEN
 11         f := 1;
 12     ELSE
 13         f := x * fact(x-1);
 14     END IF;
 15 RETURN f;
 16 END;
 17
 18 BEGIN
 19     num:= 6;
 20     factorial := fact(num);
 21     dbms_output.put_line(' Factorial ' || num || ' is ' || factorial);
 22 END;
 23 /

PL/SQL procedure successfully completed.
```

```
SQL> create function fnfact(n number)
  2 return number is
  3 b number;
  4 begin
  5 b:=1;
  6 for i in 1..n
  7 loop
  8 b:=b*i;
  9 end loop;
10 return b;
11 end;
12 /
```

Function created.

```
SQL> create function fnfact(n number)
  2  return number is
  3  b number;
  4  begin
  5  b:=1;
  6  for i in 1..n
  7  loop
  8  b:=b*i;
  9  end loop;
 10  return b;
 11  end;
 12  /
create function fnfact(n number)
```

**RESULT: FUNCTIONS IN SQL WERE SUCCESSFULLY  
EXECUTED**

**CHITRALEKHA.CH**

**RA1911003010387**