Ex.No. 10

PL/SQL Functions

Aim:

To manipulate Functions in PL/SQL.

PL/SQL Function

A PL/SQL **function** is a reusable program unit stored as a schema object in the Oracle Database. A function is **same as a procedure except that it returns a value.** We can call the function anywhere within its scope inside the program, such as in Boolean expressions, assignment statements to assign the return value of the function to some variable or even inside the SQL statement, and queries such as inside where clause, etc.

The following illustrates the basic syntax of creating a function in PL/SQL:

```
CREATE [OR REPLACE] FUNCTION <function_name>
  ( (<variable name> IN <datatype>,
<variable name> IN <datatype>,…)

  RETURN <datatype>

IS | AS

variable/constant declaration;
BEGIN
-- PL/SQL subprogram body;
EXCEPTION
-- Exception Handling block ;
END <procedure_name>;
```

where

- OR REPLACE specifies the function that is to replace an existing function if present.
- type specifies the PL/SQL type of the parameter.
- The body of a function must return a value of the PL/SQL type specified in the RETURN clause.

The function header has the function name and a RETURN clause that specifies the datatype of the returned value.

The function body is the same as the procedure body which has three sections: declarative section, executable section, and exception-handling section.

- The declarative section is between the IS and BEGIN keywords. It is where you declare variables, constants, cursors, and user-defined types.

- The executable section is between the BEGIN and END keywords. It is where you place the executable statements. Unlike a procedure, you must have at least one RETURN statement in the executable statement.
- The exception-handling section is where you put the exception handler code.

In these three sections, only the executable section is required, the others are optional.

Q1) Function to add Two numbers

```
create or replace function adder(n1 in number, n2 in number)
return number
is
n3 number(8);
begin
n3 :=n1+n2;
return n3;
end;
/
```

PL/SQL Block that invokes the function `adder`

```
DECLARE
  n3 number(2);
BEGIN
  n3 := adder(11,22);
  dbms_output.put_line('Addition is: ' || n3);
END;
/
```

Output:

Addition is: 33

Statement processed.

0.05 seconds

Q2) Write a PL/SQL function to find factorial of a number.

```
SQL> create or replace function fact(n number)
return number is
i number(10);
f number:=1;
begin
for i in 1..N loop
f:=f*i;
end loop;
```

```
return f;
end;
 /
```

Q3) Write a PL/SQL function to check whether given number is prime or not. Return boolean value.

Q4) Write a PL/SQL function to display the salary of the employee from emp table by getting empid as input.