Vel Tech Rangarajan Dr. Sagunthala R&D Institute of Science and Technology
(Deemed to be University Estd. u/s 3 of UGC Act, 1956)
School of Computing
B.Tech. – Computer Science and Engineering

VTR UGE2021- (CBCS)

Academic Year: 2025–2026

SUMMER SEMESTER - SS2526

Course Code : 10211CS207

Course Name : Database Management Systems

Slot No : S4-L5

DBMS TASK - 6B REPORT

Submitted by:

VTUNO	REGISTER NUMBER	STUDENT NAME
VTU30200	24UECS1453	GANESWARA REDDY UPPALAPALLI

ABSTRACT

The objective of this task is to study and implement various PL/SQL control structures, procedures, and functions to perform conditional, iterative, and modular programming inside Oracle.

PL/SQL enhances SQL with procedural capabilities such as conditional branching using IF, iterative loops (FOR, WHILE, LOOP), and modular constructs (PROCEDURE and FUNCTION).

This exercise demonstrates the use of control structures like IF - THEN, CASE - WHEN, GOTO, NULL, FOR loops, and REVERSE loops, along with examples of creating and executing stored procedures and functions.

PROCEDURES

SQL>set server output on;
SQL>create table ar(eno number(10) primary key , ename varchar(10),exp number(10),sal number(10);

Table created.

SQL>insert into ar values(10,'hema',12,30000);

1 row created.

SQL>insert into ar values(11,'kin',6,22000);

1 row created.

Example 1:

Procedure using IN parameter

SQL>create or replace procedure pro(ex in number)is sala number(10);
2 begin

```
3  select sal into sala from ar where exp=ex;
4  if sala>10000 then
5  update ar set sal=sal*0.5 where exp=ex;
6  else
7  update ar set sal=sal*1.5 where exp=ex;
8  end if;
9  dbms_output.put_line('table updated');
10  end;
11  /
```

Procedure created.

SQL>exec pro(6);
Table updated.
PL/SQL procedure successfully completed.
SQL>select * from ar;

ENO	ENAME	EXP	SAL
10	hema	12	30000
11	kin	6	50000

Example 2:

Procedure using OUT parameter

```
SQL>create or replace procedure pro1(b out number)is
2  begin
3  select eno into b from ar where exp=6;
4  end;
5  /
```

Procedure created.

```
SQL>declare
2  b number;
3  name varchar(10);
4  begin
5  pro1(b);
6  dbms_output.put_line(b);
7  select ename into name from ar where eno=b;
8  dbms_ output.put_line(name);
9  end;
10  /
```

Output:

```
11
Kin
```

PL/SQL procedure successfully completed.

FUNCTIONS

Example 1:

Create a function to find the factorial of the given numbers:

```
SQL>create or replace function fac(n number)return number is
2  i number;
3  f number;
4  begin
5  i:=1;
6  f:=1;
7  loop
8  f:=f*i;
9  i:=i+1;
10  exit when i>n;
11  end loop;
12  return f;
13  end;
14  /

SQL>declare
2  r number;
3  begin
4  r:=fac(5);
5  dbms_output.put_line('THE FACTORIAL OF THE GIVEN NUMBER IS'||r);
6  end;
7  /
```

Output:

THE FACTORIAL OF THE GIVEN NUMBER IS 120.

Example 2:

```
SQL>create or replace function fn(id number) return number is
  2      amt number;
  3       begin
  4       select sal into amt from emp where empno=id;
  5       return amt;
  6       end;
  7       /

SQL> set server output on
SQL>declare
  2      n number;
  3      begin
  4      n=fn(101);
  5      dbms_output.put_line('sal'||n);
  6      end;
  7        /
```

Output:

```
1000


PL/SQL procedure successfully completed.
```