Task-No : 7
Date : 23/9/25

PL/SQL Procedures & Functions

Aim: To implement PL/SQL Procedures, Functions and Loops on Number theory and business scenarios.

Procedure:

PL/SQL is a combination of SQL along with the procedural features of programming languages. It was developed by Oracle Corporation in the early 90's to enhance the capabilities of SQL. PL/SQL is one, three key programming languages embedded in oracle database, along with SQL it self and Java.

Simple program to print a sentence:

Syntax:-

```
DECLARE
   <declaration section>
BEGIN
   <executable section>
EXCEPTION
      < exception handling>.
END;
```

Program:

```
DECLARE
   message varchar2 (20) := 'booking closed';
BEGIN.
   dbms_output.put_line (message);
END;
```

Dynamic Input:

```
set serveroutput on;
declare
      x number (5);
      y number (5);
      z number (9);
```

```
begin
   x: =10;
   y: = 12;
   z: = x+y;
   dbms_output.put_line ('sum is' ||z);
end;
/
```

output: sum is 22.

```
declare
Var1 integer;
Var2 integer;
Var3 integer;
begin.
  Var1 := &Var1;
  Var2 := &Var2;
  Var3 := Var1 + Var2;
  dbms_output. put_line (Var3);
end;
/.
Enter value for Var1 : 20
 old 6 : Var1 := &Var1;
new 6 : Var1 := 20;
Enter value for Var2 : 30.
 old 7 : Var2 := &Var2;
 new 7 : Var2 := 30;
 50.
Declare
  Lid number(3) := 100;
Begin.
    if (Lid=10) then.
       dbms_output.put_line. ('Value of Lid is 10');
   elseif (Lid = 20) then.
       dbms_output.put_line('Value of Lid is 20');
```

```
    else if (Lid = 30) then
        dbms_output.put_line ('value of Lid is 30');
    else
        dbms_output.put_line ('None of the values is matching');
    End if;
    dbms_output.put_line ('Exact value of Lid is : '||Lid);
End;
/
```

Output:

```
None of the values is matching
Exact value of Lid is : 100.
```

```
Declare
    Lid number(1);
    Old number(1);
Begin
    <<outer_loop>>
    For Lid in 1...3 loop
        <<Inner_loop>>
        For oid in 1...3 loop
            dbms_output.put_line('Lid is : '||Lid||' and oid is : '||oid);
    END loop Inner_loop;
    END loop outer_loop;
END;
/
```

```
Lid is : 1 and oid is : 1
Lid is : 1 and oid is : 2
Lid is : 1 and oid is : 3
Lid is : 2 and oid is : 1
Lid is : 2 and oid is : 2
Lid is : 2 and oid is : 3
Lid is : 3 and oid is : 1.
Lid is : 3 and oid is : 2
Lid is : 3 and oid is : 3
```

Program for only procedure:

```
create or replace procedure as information
(C-id in number, C-name in varchar2)
is
begin
dbms_output.put_line ('ID:'||c-id);
dbms_output.put_line ('Name:'||c-name);
end;
/
```

procedure created

```
exec as information (101, 'raam');
```

PL/SQL procedure successfully completed

```
set serveroutput on;
exec as information (101, 'raam');
```

ID: 101
Name: raam.

Program for only functions.

```
create or replace function asinformation.
(ch-id in number c-name in varchar2)
Return varchar2
Is
begin
if (c-id > 200 then)
Return ('no booking available');
False
Return ('booking open');
End if;
End;
```

Function created.

```
declare
msg varchar2(200);
begin
msg:= csinformation2(102, 'raam');
dbms_output.put_line(msg);
end;
/
```

Vehicle_available.

```
declare
msg varchar2(200);
begin
msg=csinformation2(206, 'raam');
dbms_output.put_line(msg);
end;
/
```

No vehicle available.

VEL TECH	
EX No.	7
PERFORMANCE (5)	5
RESULT AND ANALYSIS (5)	5
VIVA VOCE (3)	3
RECORD (4)	
TOTAL (15)	15
SIGN WITH DATE	

Result: Thus the implementation of PL/SQL procedures, functions and loops on number theory and business scenario was completed

Task No: 7B

Aim: To implement PL/SQL programs using loops for printing prime number customer IDs and for demonstrating loop control in different scenario.

Procedure:

1. Start a PL/SQL block or procedure.
2. Use a cursor (if required) to fetch customer IDs from a table.
3. For each ID, check whether it is a prime number using a loop.
4. Use for loop / while loop to demonstrate prime number checking
5. Print the result using dbms_output.put_line.
6. End the block.

Using while loop with cursor.

```
Create or replace procedure print_prime customer IS.
   cursor cust_cur is
      select customer_id from customers;
   v_id number;
   v_is_prime. Boolean:
    v_i Number;
Begin.
   open_cust_cur;
   loop
    fetch cust_cur into v_id;
    exit when cust_cur%. Not found;
    If v_id <2 then.
        v_is_ prime := False;
    False
        v_is_prime := True;
        v_i := 2;
        while v_i <= True (sort (v_id)) loop.
          If MOD (v_id, v_i) = 0 THEN
             v_is_prime := FALSE;
             EXIT;
          END IF;
```

```
        V_i := V_i+1;
      END loop;
    END IF;
  If V_is_prime THEN
    DBMS_OUTPUT.PUT_LINE('prime customer ID:'||V_id);
  END If;
 END loop;
 CLOSE_CUST_CUR;
END;
```

Using for loop for First N prime Numbers.

```
create or Replace procedure print_first_n_primes(n NUMBER)IS
  V_num NUMBER := 2;
  V_cust Number := 0;
  V_is_prime Boolean;
Begin
   while V_count < n loop.
   V_is_prime := TRUE;
   FOR I IN 2.. TRUNC (SQRT (V_num)) loop.
     IF MOD (V_num, i) = 0 THEN
       V_is_prime := FALSE
       EXIT;
     END IF;
   END loop;
   If V_is_prime THEN
     DBMS_output.put_line ('prime :'||V_num);
     V_count := V_count + 1;
     END IF;
     V_num := V_num + 1;
   END loop;
END;
```

EX No.	15
PERFORMANCE (5)	5
RESULT AND ANALYSIS (5)	5
VIVA VOCE (3)	5
RECORD (4)	
TOTAL (15)	15
SIGN WITH DATE	

Result: Thus the Implementation of PL/SQL procedures, functions and loops on number theory was completed.