Task-7 PL/SQL procedure for LOOPS

Aim:- TO implement PL/SQL procedures, functions and loops on Number theory and business Scenarios.

procedure:-

PL/SQL is combination of SQL along with the procedual features of programming languages. It was developed by oracle corporation in the early 90s to enhance the capabilities of SQL. PL/SQL is one of three key programming languages embedded in the in the oracle database, along with SQL itself and Java.

Section of Description

S.NO	Declarations
1.	This section starts with the keyword DECLARE. It is an optional section and defines all variables, Cursors, Subprogram, and another elements to be used in program
2.	Executable Commands This section is enclosed between keywords BEGIN and END and it is a mandotory section. It consists of execute PL/SQL
3.	Exception Handling This section starts with the keyword EXCEPTION. This optional section contains exceptions(s) the handlee errors in program.

Simple program to print a sentence

Syntax:

```
DECLARE
<declaration section>
BEGIN
<executable commands(s)>
EXCEPTION
<exception handling>
END;
```

Program :

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

Static Input

```
SQL> set serveroutput on
SQL> declare
  2  x number(5);
  3  y number(5);
  4  z number(9);
  5   begin
  6   x:=10;
  7   y:=12;
  8   z:=x+y;
  9. dbms_output_put_line('sum is'::z);
 10. end;
 11  /
  Sum is 22
```

PL/SQL procedure Successfully completed.

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_line('sum is'||z);
end;
/
```

```
SQL>declare
  2  var1 Integer;
  3.  var2 integer;
  4.  var3 integer;
  5   begin
  6   var1:=&var1;
  7   var2:=&var2;
  8   var3:=var1+var2;
  9   dbms_output.put_line('var3');
 10   end;
 11   /
```

Enter value for var 1:20
old 6: var1:=&var1;
new 6:var1:
Enter value for var 2:30
old 7:var2:&var2;
new 7:var2:=30;

PL/SQL procedure Successfully Completed.

```
DECLARE
   hid number (3):=100;
BEGIN
   IF (hid=10) THEN
   dbms_output.put_line('value of hid is 10');
   ELSEIF (hid=20) THEN
   dbms_output.put_line('value of hid is 20');
   ELSEIF(hid=30) THEN
   dbms_output.put_line('value of hid is 30');
   ELSE
   dbms_output.put_line('None of the values is
                         matching');
   END IF;
```

```
dbms_output.put_line('exact value of hid is:'|| hid);
END;
/
```

None of the values is matching
exact values of hid is :100
PL/SQL procedure successfully completed.

```
DECLARE
    hid number(1);
    oid number(1);
BEGIN
  <<outer loop>>
  FOR hid IN 1...3 LOOP
    dbms_output.put_line 'hid is:' || hid || 'and oid
is:' || oid);
    END loop inner_loop;
    END loop outer_loop;
    END;
```

hid is: 1 and oid is: 1
hid is: 1 and oid is: 2
hid is: 1 and oid is: 3
hid is 2 and oid is: 1
hid is 2 and oid is: 2
hid is 2 and oid is: 3
hid is 3 and oid is: 1
hid is 3 and oid is: 2
hid is 3 and oid is: 3

PL/SQL procedure successfully completed.

Sample program for only procedure:

```
SQL> create or replace procedure csinformation
  2. (c_id in number, c_name in varchar2)
  3 is
  4 begin
  5 dbms_output.put_line('ID: '||c_id);
  6 dbms_output.put_line('Name: '||c_name);
  7 end;
  8 /
Procedure created.
SQL> exec csinformation(101,'raam');
PL/SQL procedure successfully completed.

SQL> set serveroutput on;
SQL> exec cinformation(101,'raam');
Name: raam
PL/SQL procedure successfully completed.
```

Sample Program for only function:

```
SQL> create or Replace function csinformation
(c_id in number, c_name in varchar2)
Return varchar2
IS
Begin
If c_id > 200 then
Return('no booking available');
Else
Return('booking open');
End if;
End;
```

Function created

```
SQL> declare
  2 mesg varchar2(200);
  3 begin
  4 mesg:=csinformation2(102,'raam');
  5 dbms_output.put_line(mesg);
  6 end;
  7 /
```

vehicle available

```
SQL> declare
  2 mesg varchar2(200);
  3 begin
  4 mesg:=csinformation2(206,'raam');
  5 dbms_output.put_line(mesg);
  6 end;
  7 /
```

NO vehicle available

PL/SQL procedure Successfully Completed.

Result: Thus, the ~~program~~ implementation of PL/SQL procedure functions and loops an number theory and business scenario has been executed successfully.

Task:- 7.1

PL/SQL procedure for LOOPS.

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

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_LIN
6. End the block.

Example:- 1 .Using WHILE LOOP with cursor

prime check using WHILE loop.

```
CREATE OR REPLACE PROCEDURE print_first_n
_primes (n NUMBER) IS
    v_num NUMBER := 2;
    v_count NUMBER := 0;
    v_is_prime BOOLEAN;
    BEGIN
WHILE v_count < n LOOP
 v_is_prime := TRUE;
 for i IN 2 .. FLOOR (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;
/
```

This procedure checks all customer ID's in the table and points the prime ones using a WHILE LOOP.

EXAMPLE :- 2 : Using for LOOP for first N prime number.

```
CREATE OR REPLACE PROCEDURE Print_first_n_
primes (n Number) IS
    v_num Number := 2;
    v_count 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;
```

This procedure prints the first N prime Numbers using For Loop.

```
BEGIN
    print_first_n_primes(10);
END;
```

This procedure prints the First N prime Numbers.

VEL TECH	
EX NO.	7
PERFORMANCE (5)	5
RESULT AND ANALYSIS (5)	5
VIVA VOCE (5)	5
RECORD (5)	/
TOTAL (20)	15

Result: Thus, Implementation of PL/SQL procedures functions and Loops on number theory has been successfully executed.