22/09/25 Task-7-PL/SQL Procedures, functions, loops

Aim:- To implement PL/SQL Procedures, functions and loops on number theory and bussiness scenarios

Procedure:-

PL/SQL is combination of SQL along with procedural features of programming languages. it was developed by oracle corporation in the early 90's to enhance the compabilities of SQL. PL/SQLs is one of three key Programming languages embedded in oracle database, along with SQL itself and java.

Declarations:-

This section starts with the keyword Declare it is an optional section and defines all variables cursors, subprograms and other elements to be used in the program

Executable commands:-

This section is blw BEGIN and END and it is mandatory sections. it is should have at least one executable code line which may be just a NULL command to indicate that nothing should be executed

Exception Handling:-

This section starts with keyword Exception

syntax:-

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

Query:-

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

Query:-

```
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:-

Addition of x & y sum is 22

Query:-

```
SQL>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;
```

output:-

50

Query:-

```
hid number(3) := 100;
BEGIN
    if (hid =10) then
      dbms_ouput.Putline ('value of hid is 10');
    Else if (hid=20) then
      dbms_output.Put_line ('value of hid is 20');
```

```
Else if (hid = 30) Then
    dbms_output.put_line ('value of hid is 30');
        Else
    dbms output.put_line ('None of the values
                    is matching');
        ENDIF;
    dbms output.put_line('Exact value of hid is '||hid);
    END;
    /
```

output:-

None of the value is matching
Exact value of hid is: 100

loop:-

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

Query

```
DECLARE
    hid number (1);
    hid number (1);
BEGIN
    <<outer loop>>
```

```
for hid in 1..3 loop
  <<inner loop>>
  for bid in 1..3 loop
  dbms_output.put_line ('hid is: '||hid||' and
      oid is '||oid);
      END loop inner-loop;
      END loop outer-loop;
      END;
      /
```

Program for procedure:-

```
SQL>create or replace procedure cs 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
  SQL>exec cs information (101,'raam');
  PL/SQL Procedure successfully completed.
  SQL> set serveroutput on;
  SQL> exec csinformation (101,'raam');
  ID:101
  Name: raam
```

Program for only function

```
SQL>create or replace function csinformation
(h_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> created~~

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

vechicle available

SQL > declare

```
mesg varchar2 (200);
begin
mesg := cs information 2 (206, 'ram');
dbms - output Put - line (mesg);
end;
/
```

No vechicle available

For while loop

```
create or replace Procedure Print_Prime_customers 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;
        Else
            v_is_Prime := True;
            v_i := 2;
        While v_i <= Trunc (sqrt (v_id)) Loop
        if Mod (v_id, v_i) = 0 Then
        v_is_prime := - false;
        Exit;
        Endif;
        v_i := v_i + 1;
```

```
End loop;
Endif
If v_is_Prime then
    DBMS_output.Put_line('Prime customer ID:' || v_id);
    End if;
  End loop;
  close cust_cur;
End;
```

For 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_Count := 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;
    Endloop;
End;
```

VEL TECH - C...	
EX.NO.	7
PERFORMANCE (5)	6
RESULT AND ANALYSIS (5)	5
VIVA VOCE (5)	5
RECORD (5)	-
TOTAL (20)	16
SIGN WITH DATE	

22/9/24

Example:-
```
Begin
  Print first_n_Prims (10);
End;
```

Result:- Thus the PL/SQL Programs using loops for prime number is executed successfully