Task no: 7(a) PL/SQL Procedures and functions.
Date: 23/9/25

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, there key programing languages embeddad in oracle database along with SQL itself and Java.

Simple problem to print a sentence.

Syntax:

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

Program:

```
DECLARE
    MESSage varchar2(20) := 'hoting 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;

SO

```
DECLARE
    hid Number(3) := 100;
BEGIN
  if (hid = 10) then
     dbms_output.put_line('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');
  end if;
  dbms_output.put_line('Exact value of hid
                       is :' ||hid);
  END;
  /
```

Output:

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

```
DECLARE
      hid number(1);
      Old number(1);
BEGIN
      <<outer_loop>>
      for hid in 1 ... 3 loop
         <<inner_loop>>
         for oid 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 no : 1 and oid no : 1
hid no : 1 and oid no : 2
hid no : 1 and oid no : 3
hid no : 2 and oid no : 1
hid no : 2 and oid no : 2
hid no : 2 and oid no : 3
hid no : 3 and oid no : 1
hid no : 3 and oid no : 2
hid no : 3 and oid no : 3

Program for only Procedure

```
create or replace procedure csinformation
(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 csinformation(101,'ram');
```

PL/SQL procedure successfully completed

```
Set serveroutput on;
exec csinformation(101,'ram');
```

ID:101
Name: ram

Program for only functions.

```
Create or replace function csinformation
(ch_id in number, c_name in varchar2)
return varchar2
is
Begin
if(id>200 then
Return ('no booking available');
else
```

return ('booking open');
END 90;
END;

function created

Declare
msg := csinformation2(102, 'ram');
dbms_output.put_line(msg);
end;
/

output: booking open

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

output: no booking available.

Task: 7(b)
Date: 23-07-25

PL/SQL Loops

Procedure:

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

Using while loop with cursor.

```
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%NOTFOUND;
        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;
        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_curi
  END;
```

Procedure closed.

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;
```

Procedure created.

```
Ex: Begin
      Print_First_n_Primes(10);
    End;
```

output:

Prime: 2
Prime: 3
Prime: 5
Prime: 7
Prime: 11
Prime: 13
Prime: 17
Prime: 19
Prime: 23
Prime: 29

VEL TECH	
EX NO.	7
PERFORMANCE (5)	5
RESULT AND ANALYSIS (5)	5
VIVA VOCE (5)	5
RECORD (5)	
TOTAL (20)	18
SIGN WITH DATE	

23/9/25

Result: The PL/SQL procedure functions and loops were successfully implemented.