

## Task 2:- PLI SQL Procedure function and logic.

Aim:- to implement PLISQL procedure, functions and loops.

Sample PLISQL Program (static input):-

```
DECLARE
    message VARCHAR2(20); := 'Booking closed'
BEGIN
    dbms_output.put_line(message);
END;
```

Output:-

Booking closed

conditional statement (dynamic input);

```
DECLARE
    nId. 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('Name of the value is matching');
    END IF;
    dbms_output.put_line('Exact value of nId is: ' || nId);
END;
```

output:-

None of the value is matching  
exact value of hid is 100

3. Nested loops example:-

DECLARE

hid Number (1);

vid Number (1);

BEGIN

ccouted - loop >>

for vid IN 1....x DO

dooms - out put - put - line ('hid is :||hid||  
and vid is "||vid||')

END LOOP inner - loop;

END LOOP outer - loop;

END;

output:- hid is :1 and vid is = 1

hid is :1 and vid is = 2

hid is :1 and vid is = 3

hid is :2 and vid is = 1

~~hid is :2 and vid is = 2~~

hid is :2 and vid is = 3

hid is :3 and vid is = 1

hid is :3 and vid is = 2

hid is :3 and vid is = 3

4 Procedure example

CREATE OR REPLACE PROCEDURE  
booking - store (c <= id FN number)



```

IS
BEGIN
  IF (-Id) < 200 THEN
    dbms_output.put_line('No working
    available');
  ELSE
    dbms_output.put_line('Booking open');
  END;
/

```

```

Execution:
BEGIN
  booking - status (150);
  booking - status (250);
END;
/

```

out put:-

Booking open  
no booking available.

PLI SQL procedure for loop.

Example:- using while loop with cursor  
from check using WHILE loops for  
patient IDs.

~~DECLARE~~

```

CURSOR Pat - cur is
  SELECT patient - FROM Patient;
  ( - id Patient Patient - id, TYPE
  i number;
  flag number;

```

```

BEGIN
  OPEN Pat - cur;
  FETCH Pat - cur INTO op - id;

```

WHILE pat - cur = 1. Found LOOP

```
flag := 0;  
for N 2. -- P_id(2) LOOP  
  if MOD(P_id - i) = 0 THEN  
    flag := 1;  
  exit.
```

END IF;

END LOOP;

Example 2:- using for loop for first N prime  
Patient IDs

DECLARE

n number = 10;

count number := 0;

i number := 2;

j number;

flag number;

BEGIN

WHILE COUNT < n LOOP

flag := 0

FOR j < n 2. -- i/2 LOOP

if MOD(i, j) = 0 THEN

flag := 1;

exit

END IF;

END LOOP;

if flag = 0 THEN

dbms\_output.put\_line('prime

patient = 0: 11);

count := count + 1;

```

i = Number; i = 2;
j = Number;
*
END IF;
i := i + 1;
END LOOP;
END;

```

VEL TECH - CSE	
EX NO.	
PERFORMANCE (5)	07
RESULT AND ANALYSIS (5)	5
VIVA VOCE (5)	5
RECORD (5)	5
TOTAL (20)	15
SIGN WITH DATE	15

22/9/25

Result shows the all functions and loops executed successfully.