Task 7: PL/SQL Procedure Function and logic

Aim: to implement PL/SQL procedure, Functions and loops.

Sample PL/SQL Program (static input):

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

Output:

```
Booking closed
```

conditional statement (Dynamic input):

```
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
                         value is matching');
      END IF;
      dbms_output.put_line('Exact value of
                         hid is:'||hid);
END;
```

output:

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

3. Nested Loops Example:

```
DECLARE
    hid NUMBER(1);
    oid NUMBER(1);
BEGIN
    <<outer_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;
/
```

output:

```
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
```

4. Procedure Example

```
CREATE OR REPLACE PROCEDURE booking_store
                         (c_id IN NUMBER)
IS
BEGIN
    IF (c_id)>200 THEN
    dbms_output.put_line('no booking available');
```

```
ELSE
   dbms_output.put_line('Booking open');
 END;
/
```

Execution:

```
BEGIN
   booking_status(150);
   booking_status(250);
 END;
/
```

output:

```
Booking open
NO booking available
```

PL/SQL procedure for loop

Example 1: using WHILE LOOP with cursor
prime check using WHILE LOOP for
patient IDs

```
DECLARE
     CURSOR pat_cur is
        SELECT patient_id FROM patient;
     p_id patient.patient_id%TYPE;
     i NUMBER;
     flag NUMBER;
BEGIN
   OPEN pat_cur;
   FETCH pat_cur INTO p_id;
   WHILE pat_cur%FOUND LOOP
        flag:=0;
   FOR i IN 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  NUMER:=2;
    j  NUMBER;
    flag NUMBER;
  BEGIN
   WHILE count <n LOOP
   flag:=0
     FOR j IN 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 ID:'||i);
     COUNT:=COUNT+1;
   END IF;
   i:=i+1;
   END LOOP;
   END;
```

22/9/25

Result:
Thus the PL SQL Functions and loops executed successfully.

VEL TECH - CSE	
EX NO.	7
PERFORMANCE (5)	5
RESULT AND ANALYSIS (5)	5
VIVA VOCE (5)	5
RECORD (5)	—
TOTAL (20)	15
SIGN WITH DATE	[illegible]