

7.	22/09/25	Procedure Function and loops: Program using PL/SQL procedures, functions and loops	VS	V <del>(re)</del> <del>2018/19</del>
----	----------	--	----	--

22/9/25 Task-7: Procedure Function and loops : Program  
Using PL/SQL Procedures, Functions & loops

Aim:-

To implement PL/SQL procedures, functions and loops on number theory and business scenarios.

1. Simple PL/SQL program (Static Input)

DECLARE

Message VARCHAR2(20) := 'Booking Closed'

BEGIN

dbms\_output.put\_line(Message);

END;

Output:

Booking closed

2. Conditional Statement (Dynamic Input):

DECLARE

hid NUMBER(3) := 100;

BEGIN

IF &(hid = 10) THEN

dbms\_output.put\_line('Value of hid is 10');

ELSIF (hid = 20) THEN

dbms\_output.put\_line('Value of hid is 20');

ELSIF (hid = 30) THEN

dbms\_output.put\_line('Value of hid is 30');

ELSE

dbms\_output.put\_line('None of the values 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);

Did NUMBER(1);

```

BEGIN
  << outer-loop >>
  for hid IN 1..3 loop
    << inner-loops >>
    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_status
  (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 IF;

END;

Execution

BEGIN

booking\_status(150);

booking\_status(250);

END;

Output:

Booking open

No booking available

### PL/SQL Procedure for Loops

Example 1: Using WHILE Loop with Cursor

Prime check using 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 THEN 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;
```

```
            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_Cur;
```

```
END;
```

The procedure checks all Customer IDs in the table and prints the prime ones using a WHILE LOOP.

Example 2: Using for Loop for first N prime Numbers.

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 (SQR (v-num)) Loop

If MOD (v-num) = 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 a FOR LOOP.

BEGIN

Print-first-n-primes (10);

END;

QUESTION NO. 1: Write a PL/SQL program to calculate the total marks obtained by a student in five subjects. The subjects are Performance, Result and Analysis, Viva Voce, Record and Total. The marks for each subject are stored in separate variables. The total marks are calculated by adding all the individual marks. The final result is displayed on the screen.

VEL TECH	
EN No.	7
PERFORMANCE (5)	6
RESULT AND ANALYSIS (5)	6
VIVA VOCE (5)	5
RECORD (5)	
TOTAL (20)	16
SIGN WITH DATE	C 22/10/13

Result:-

Thus, the procedure function and loops program using PL/SQL procedure, functions and loops are executed successfully.