22.9.25

TASK -7
PL/SQL

Aim: To implement PL/SQL Procedures, Functions and loops on Number theory and buisness sunarios.

Procedure : PL/SQL is a combination of SQL along with procedure features of Programing languages. It was developed by oracle corporation in the early 90's to enhance the capabilities of SQL. PL/SQL is one of three key programing languages embedded in the oracle Database, along with SQL itself and Java.

1.) simple PL/SQL Program (static input)

```
DECLARE
    Message VARCHAR (20) = "Booking closed';
BEGIN
    dbms_output_put_line (message);
END
```

output : Booking closed.

2.) Conditional statement (Dynamic Input)

```
DECLARE
        sid NUMBER (2) = 100;
BEGIN
        if (sid = 10) THEN
  dbms_output_putline ('value of hid is 10');
  ELSE IF (sid = 20) THEN
        dbms_output_put line ('value of hid is 20');
  ELSE IF (sid = 30) THEN
        dbms_output_put line ('value of hid is 30');
  END IF
        dbms_output_put line ('exact value of hid (hid));
END;
```

output :

```
None of the value is matching
Exact Value of hid 10 = 10
```

3.) Nested loops Example :

```
DECLARE
        hid NUMBER (1);
        hid NUMBER (1);

BEGIN
        << outer loop >>
      for hid (1) ... loot
          << inner loop >>
      for hid (3) 1.... loop
            dbms_ output_ put line
   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
<cid NUMBER>
IS
BEGIN:
 If (cid >200 THEN
      dbms_output_put_line ('No booking available')
ELSE
      dbms_output_line ('Booking open');
ENDIF:
```

execution :

```
BEGIN
    Booking_status(130);
    Booking_status(250);
END;
```

output :

```
Booking open
No Booking available
```

FoR PL/SQL Procedure for loops

Example 1: using WHILE loop with cursor

```
 CREATE OR REPLACE PROCEDURE PRINT_Prime_Customer ID
 Cursor cust_CUR IS
       SELECT CUSTOMER_id from customer;

   V-id NUMBER;
   v_is Prime BOOLEAN;
   v_i NUMBER;

  BEGIN
       open cust-cur;
       Loop
       FETCH cost-cur INTO v-id;
 EXIT THEN COST_CUR;
  If V-id < 2 THEN
      V_i := 2;
     WHILE V_i<=TRUWL(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;
       ENDSE cost-cur;
       END;
```

The Procedure checks are customer Ios in table and prints the prime ones using a loop.

Example 2: using for loop for first N prime numbers

```
     CREATE OR REPLACE PROCEDURE PRINT FIRST_n_Primes in numy
      v_num NUMBER := 2
```

```
v_count NUMBER : = 0
v_is_Prime BOOLEAN ;

BEGIN
  WHILE v_count = n loop
  v_is print ;= True ;
  FOR i IN 2 TRUNCATE (SORT ( vnum)) Loop
        of MOD(v_num) = 0 THEN
         v_is_Prime := FALSE ;
            EXIT ;
     END IF ;
     END LOOP ;
      of v_is_Prime THEN
          DBMS_OUTPUT.PUT_LINE ('PRIME :' ||v_num) ;
       v_count : = v_count +1
    END IF ;
    END LOOP ;
   END ;
```

The Procedure Prints the first n Prime numbers using a for loop.

VEL TECH - CSE	
E* NO.	07
RFORMANCE (5)	5
ULT AND ANALYSIS (5)	5
VOCE (5)	5
	15

Result: Thus Procedure function, loops program using PL/SQL and loops are executed succesfully.