

Task-7(1)

PL/SQL procedures ,Function · loops.

Aim: To implement PL/SQL procedures ,function · and loops on number theory and business scenarios.

procedures

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 in one of three key programming languages embedded in the oracle Database along with SQL itself and Java.

Selection of Description.

5.NO Declarations.

1 This section starts with the keyword DECLARE . It is an optional section and defines all variables , cursors , subprogram , and another elements stored used in program.

2 Executable commands

This section is enclosed b/w keyword BEGIN and END .
it is a mandatory section . It consists of execute PL/SQL

3 Exception Handling

This section starts with the keyword EXCEPTION this optional section contains · exceptions (g) the hundred error in program.

Simple program to print a sentence.

Syntax:

```

DECLARE
<declarations section>
BEGIN
<executable commands>
EXCEPTION
<exception handling>
END;

```

program:

```

DECLARE
message varchar2(20):= 'booking closed';

```

```
BEGIN  
    dbms_output_line(message);  
END;
```

static input:

SQL> set server output on

SQL> declare

```
2  x number<5>;  
3  y number<5>;  
4  z number<9>;  
5  begin  
6  x:=10;  
7  y:=12;  
8  z:=x+y;  
9  dbms_output_line('sum is'||z);  
10 end;  
11 /
```

sum is 2.

PL/SQL procedure successfully completed

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_line('sum is '||z);  
end;
```

SQL> declare

```
2  var1 integer;  
3  var2 integer;  
4  var3 integer;  
5  begin  
6  var1 :=&var1;  
7  var2 :=&var2;  
8  var3 := var1 + var2;  
9  dbms_output_line(var3);
```

10 end;

11)

Enter value for var1: 20

old '6 : var1 := &var1;

new 6: var1 := 20;

Enter value for var2: 30

old 4 : var2 := &var2;

new 4 : var2 := 30;

50

PL/SQL procedure successfully completed

DECLARE

 id number (3) := 100;

BEGIN

 IF (chid = 10) THEN

 dbms_output.put_line('value of hid is 10');

 ELSE IF (chid = 20) THEN

 dbms_output.put_line('value of hid is 20');

 ELSE IF (chid = 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('None of the values is matching');

END IF;

 dbms_output.put_line('Exact value of hid is: ' || hid);

END;

,

None of the value is matching

Exact value of hid is: 100.

PL/SQL procedure successfully completed

DECLARE

 hid number (1);

 old number(1);

BEGIN

~~< outer-loop >~~

```
FOR old IN 1...3 loop
  dbms-output.put-line 'hid is:' || hid || 'and o id is:' ||
    old};
```

END loop inner-loop;

END loop outer-loop;

END;

1

hid is:1 and o;id is:1

hid is:1 and o;id is:2

hid is:1 and o;id is:3

hid is:2 and o;id is:1

hid is:2 and o;id is:2

hid is:2 and o;id is:3

hid is:3 and o;id is:1

hid is:3 and o;id is:2

hid is:3 and o;id is:3

PL/SQL procedure successfully completed.

sample program for only procedure-

SQL> create or replace procedure is information;

2 <c>id in number , c-name in varchar 2,

3 is

4 begin

5 dbms-output -put-line <'ID : ' || <-id > ;

6 dbms-output -put-line <'Name: ' || c-name > ;

7 end;

8 /

procedure create();

SQL> exec is information (101,'raam');

PL/SQL procedure successfully completed :

SQL> set serveroutput on;

SQL> exec is information (101,'raam');

ID:101

Name:raam

PL/SQL procedure successfully completed

sample program for only function.

SQL> create or replace function csinformation

ch_id in number, c-name in varchar2;

Return varchar2

IS

Begin

If c-id > 200 then

Return ("no booking available");

else .

Return ('booking open');

End If ;

END ;

1.

function created

SQL> declare

2 mesg varchar2<200>;

3 begin

4 mesg := csinformation 2<102; room'>;

5 dbms -output .put -line <mesg>;

6 end ;

7 ,

vehicle available

SQL> declare

2 mesg varchar2<200>;

3 begin

4 mesg := csinformation 2<206; room>;

5 dbms -output .put -line <mesg>;

6 end ;

7 ,

No vehicle available.

PL/SQL procedure successfully completed.

Result: Thus, Implementation of PL/SQL procedures for loops and functions has been successfully completed.

Task 7.1

Aim: To write PL/SQL procedures for loops, prime number customer ID and for printing loop control in different scenarios.

Procedure:

- 1) Start a PL/SQL block or procedure
- 2) use a cursor to fetch customer ID's from a table
- 3) For each ID, check whether it is a prime number using a loop
- 4) use for loop/while loop to demonstrate prime number checking
- 5) print the result using DBMS_OUTPUT.PUT_LINE.

Ex + 1: using while loop with cursor.

Create OR Replace procedure print-prime customer.

~~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 NOT found;

--- prime check using while loop.

If v-is < 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: || vid);
END IF;
END LOOP;
CLOSE cust_cus;
END;

```

This procedure checks all customers IDs in the table and prints the prime one's 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(SORT(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;
```

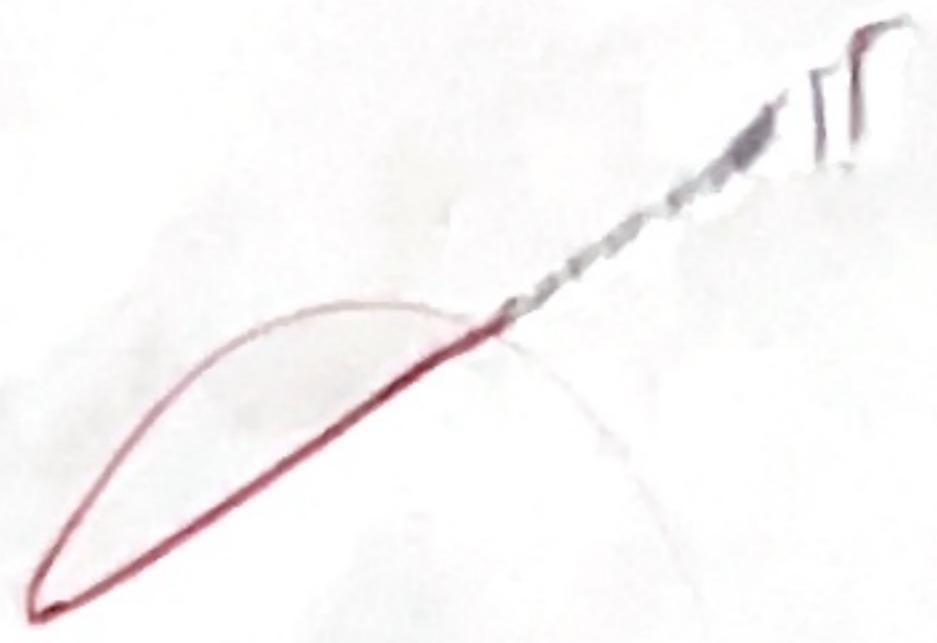
This procedure prints the first N prime numbers using
to a FOR LOOP

FOR Example:

BEGIN

```
Print-first-n-primes(10);
```

```
END;
```



VEL TECH	
EX NO.	7
PERFORMANCE (5)	6
RESULT AND ANALYS'S (5)	5
VIVA VOCE (5)	6
RECORD (5)	16
TOTAL (20)	33
WITH DATE	C 23/9/13

Result: Thus implementation of pl/sql procedures
functions and loops on number theory has
been successfully executed.