**PRACTICAL NO. 5**

**AIM:Writing PL/SQL Blocks with basic programming constructs for following Iterative Structure:**

**1. While-loop Statements**

**2. For-loop Statements.**

**1.1 Write a program to find reverse of a number entered by the user.**

**PROGRAM:**

SQL> declare

2 n number;

3 i number;

4 rev number:=0;

5 r number;

6 begin

7 n:=&n;

8 while n>0

9 loop

10 r:=mod(n,10);

11 rev:=(rev\*10)+r;

12 n:=trunc(n/10);

13 end loop;

14 dbms\_output.put\_line('Reverse is '||rev);

15 end;

16 /

**O/P:**

Enter value for n: 1234

old 7: n:=&n;

new 7: n:=1234;

Reverse is 4321

PL/SQL procedure successfully completed.

**1.2 Write a PL/SQL program to print Fibonacci series.**

**PROGRAM:**

SQL> declare

2 f1 number(3);

3 f2 number(3);

4 f3 number(3);

5 numnumber(3);

6 begin

7 f1:=0;

8 f2:=1;

9 f3:=0;

10 num:=1;

11 while num<=10

12 loop

13 dbms\_output.put\_line(f3);

14 f1 :=f2;

15 f2:=f3;

16 f3:=f1+f2;

17 num:=num+1;

18 end loop;

19 end;

20 /

**O/P:**

0

1

1

2

3

5

8

13

21

34

PL/SQL procedure successfully completed.

**2.1 Write program to find prim number between 1 to 50.**

**PROGRAM:**

SQL> DECLARE

2 counter NUMBER;

3 k NUMBER;

4 BEGIN

5 FOR n IN 1..50 LOOP

6 counter := 0;

7 k := floor(n/2);

8 FOR i IN 2..k LOOP

9 IF (mod(n, i) = 0 ) THEN

10 counter := 1;

11 END IF;

12 END LOOP;

13 IF (counter = 0) THEN

14 DBMS\_OUTPUT.PUT\_LINE(n||' is prime number');

15 END IF;

16 END LOOP;

17 END;

18 /

**O/P:**

1 is prime number

2 is prime number

3 is prime number

5 is prime number

7 is prime number

11 is prime number

13 is prime number

17 is prime number

19 is prime number

23 is prime number

29 is prime number

31 is prime number

37 is prime number

41 is prime number

43 is prime number

47 is prime number

PL/SQL procedure successfully completed.

**2.2 Write program to check wheather given number is palindrome or not.**

**PROGRAM:**

SQL> declare

2 g varchar2(20);

3 r varchar2(20);

4 i number(4);

5 begin

6 g:='&g';

7 for i in reverse 1.. length(g) loop

8 r:=r || substr(g,i,1);

9 end loop;

10 dbms\_output.put\_line('reverse string is ' || r);

11 if r=g then

12 dbms\_output.put\_line('String is Palindrome');

13 else

14 dbms\_output.put\_line('String is not Palindrome');

15 end if;

16 end;

17 /

**O/P:**

Enter value for g: nitin

old 6: g:='&g';

new 6: g:='nitin';

reverse string is nitin

String is Palindrome

PL/SQL procedure successfully completed.