Expt.No:

Date

Page No. 01

Experiment - 1

1. Write a PL/SQL code to accept the value of A, B, C and display which is greater.

```
DECLARE
  A NUMBER
  B NUMBER
  C NUMBER
BEGIN
  A := &A;
  B := &B;
  C := &C;
IF (A>B) AND (A>C) THEN
     DBMS_OUTPUT.PUT_LINE('A is greatest:' ||A);
ELSIF (B>A) AND (B>C) THEN
     DBMS_OUTPUT.PUT_LINE('B is greatest:' ||B);
ELSIF (C>B) AND (C>B) THEN
     DBMS_OUTPUT.PUT_LINE('C is greatest:' ||C);
ELSE
     DBMS_OUTPUT.PUT_LINE('TWO or more numbers
                          are equal and greatest');
END IF;
END;
```

Teacher's Signature : _____

GREATEST NUMBER.sql

```sql
SET SERVEROUTPUT ON;

DECLARE
   a NUMBER;
   b NUMBER;
   c NUMBER;
BEGIN
   -- Accept user input
   a := &a;
   b := &b;
   c := &c;

   -- Compare and display greatest number
   IF (a > b AND a > c) THEN
      DBMS_OUTPUT.PUT_LINE('A is greatest: ' || a);
   ELSIF (b > a AND b > c) THEN
      DBMS_OUTPUT.PUT_LINE('B is greatest: ' || b);
   ELSIF (c > a AND c > b) THEN
      DBMS_OUTPUT.PUT_LINE('C is greatest: ' || c);
   ELSE
      DBMS_OUTPUT.PUT_LINE('Two or more numbers are equal and greatest.');
   END IF;
END;
/
```

```
SQL> DECLARE
        a NUMBER;
        b NUMBER;
        c NUMBER;...
Show more...


B is greatest: 120


PL/SQL procedure successfully completed.

Elapsed: 00:00:00.003
```

Expt.No:

Date

Page No. 02

2. Using PL/SQL Statement, display 'welcome PL/SQL' 20 times.

```
BEGIN
    For i IN 1... 20 LooP
    DBMS_OUTPUT.PUT_LINE ('Welcome PL/SQL ');
    END LOOP;
END;
/
```

WELCOME PL/SQL.sql

```
1  SET SERVEROUTPUT ON;
2
3  BEGIN
4      FOR i IN 1..20 LOOP
5          DBMS_OUTPUT.PUT_LINE('WELCOME PL/SQL');
6      END LOOP;
7  END;
8  /
9
```

Query result | **Script output** | DBMS output | Explain Plan | SQL history

```
WELCOME PL/SQL
WELCOME PL/SQL
WELCOME PL/SQL
WELCOME PL/SQL
WELCOME PL/SQL
WELCOME PL/SQL
WELCOME PL/SQL
WELCOME PL/SQL
WELCOME PL/SQL
WELCOME PL/SQL
WELCOME PL/SQL
WELCOME PL/SQL
WELCOME PL/SQL
WELCOME PL/SQL
WELCOME PL/SQL
WELCOME PL/SQL
```

END;
/

3. Write a PL/SQL code to find factorial of a number.

```
DECLARE
    n NUMBER
    FACT NUMBER = 1;
BEGIN
    n := &n
    If n < 0 THEN
    DBMS_OUTPUT.PUT_LINE ('factorial is not defined
                          for negative numbers');
ELSE
    FOR i IN 1..n LOOP
        fact := fact * i
    END LOOP;
    DBMS_OUTPUT.PUT_LINE ('factorial of '||n||' is: '||fact);
END IF;
END;
/
```

Teacher's Signature : _____

FACTORIAL NO.sql

```
1   SET SERVEROUTPUT ON;
2
3   DECLARE
4       n NUMBER := 5;          -- Change the number here
5       fact NUMBER := 1;
6   BEGIN
7       FOR i IN 1..n LOOP
8           fact := fact * i;
9       END LOOP;
10      DBMS_OUTPUT.PUT_LINE('Factorial of ' || n || ' is: ' || fact);
11  END;
12  /
13
```

Query result | **Script output** | DBMS output | Explain Plan | SQL history

```
Elapsed: 00:00:00.005

SQL> DECLARE
        n NUMBER := 5;          -- Change the number here
        fact NUMBER := 1;
    BEGIN...
Show more...


Factorial of 5 is: 120


PL/SQL procedure successfully completed.

Elapsed: 00:00:00.006
```

Expt.No:

Date

Page No. 03

4. Write a PL/SQL code to generate fibonacci Series.

```
DECLARE
    n NUMBER;
    a NUMBER ; = 0 ;
    b NUMBER : = 1 ;
    C NUMBER ;
BEGIN
    n : = &n ;
    DBMS_OUTPUT.PUT_LINE ('Fibonacci Series upto'|| n ||'
                                                terms : ');
  If n < =0 THEN
     DBMS_OUTPUT.PUT_LINE ('Please enter a tve number');
  ELSEIF n=1 THEN
     DBMS_OUTPUT.PUT_LINE (a);
  ELSE
     DBMS_OUTPUT.PUT_LINE (a);
     DBMS_OUTPUT.PUT_LINE (b);
     FOR i IN 3.. n LOOP
            C : = a+b'
          DBMS_OUTPUT.PUT-LINE (C)
            a : =b ;
            b : =C ;
       END LOOP ;
   END IF ;
END ;
/
```

Teacher's Signature : _____

Fibonacci Series.sql

```sql
DECLARE
    n       NUMBER := 10;  -- Number of terms in Fibonacci series
    a       NUMBER := 0;   -- First term
    b       NUMBER := 1;   -- Second term
    c       NUMBER;        -- Next term
    i       NUMBER := 2;   -- Counter (starts from 2 as first two terms are f
BEGIN
    DBMS_OUTPUT.PUT_LINE('Fibonacci Series up to ' || n || ' terms:');

    -- Print first two terms
    DBMS_OUTPUT.PUT_LINE(a);
    DBMS_OUTPUT.PUT_LINE(b);

    -- Generate rest of the terms
    WHILE i < n LOOP
        c := a + b;            -- Next term
        DBMS_OUTPUT.PUT_LINE(c);
        a := b;                -- Shift values
        b := c;
        i := i + 1;
    END LOOP;
END;
/

```

```
Fibonacci Series up to 10 terms:
0
1
1
2
3
5
8
13
21
34


PL/SQL procedure successfully completed.

Elapsed: 00:00:00.003
```

Expt.No:

Date

Page No. 04

5. Write a PL/SQL code to find sum of first N no.

```
DECLARE
    n NUMBER;
    Sum NUMBER := 0;
BEGIN
    n := &n;
    FOR i IN 1..n LOOP
    Sum := Sum + i
    END LOOP;
DBMS_OUTPUT.PUT_LINE ('Sum of first' || n || ' natural
                        number is: ' || Sum );

END;
/
```

Sum of n number.sql

```sql
SET SERVEROUTPUT ON;

DECLARE
    n NUMBER := 10;        -- You can change this value
    sum_num NUMBER := 0;
BEGIN
    -- Using loop to calculate sum
    FOR i IN 1..n LOOP
        sum_num := sum_num + i;
    END LOOP;

    DBMS_OUTPUT.PUT_LINE('Sum of first ' || n || ' numbers = ' || sum_num);
END;
/
```

Query result | Script output | DBMS output | Explain Plan | SQL history

```
Elapsed: 00:00:00.003

SQL> DECLARE
         n NUMBER := 10;        -- You can change this value
         sum_num NUMBER := 0;
     BEGIN...
Show more...


Sum of first 10 numbers = 55


PL/SQL procedure successfully completed.

Elapsed: 00:00:00.006
```