Lab Experiment – 12

Title: To understand the concepts of Sequence.

Objective: Students will be able to implement the concept of sequence.

1. Create a sequence by name EMPID_SEQ starting with value 100 with an interval of 1.

```
ALTER TABLE EMPLOYEES AUTO_INCREMENT = 100;
```

2. Write a SQL command for finding the current and the next status of EMPID_SEQ.

```
SELECT LAST_INSERT_ID();
```

```
+------------------+
| LAST_INSERT_ID()  |
+------------------+
|              100 |
+------------------+
1 row in set (0.00 sec)
```

3. Change the Cache value of the sequence EMPID_SEQ to 20 and maxvalue to 1000.

```
ALTER TABLE EMPLOYEES AUTO_INCREMENT = 1000;
```

4. Insert values in the EMPLOYEES table using sequences for the Employee_id column.

```
INSERT INTO EMPLOYEES (First_Name, Last_Name, DOB, Salary, Department_id)
VALUES ('Tom', 'Hanks', '1985-12-10', 65000, 'D1');
```

5. Drop sequence EMPID_SEQ.

```
DROP TABLE EMPLOYEES;
```

Lab Experiment – 13

Title: To understand the concepts of PL/SQL programming.

Objective: Students will be able to implement the basic concepts of Pl/SQL.

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

```
DECLARE
    A NUMBER := 5;
    B NUMBER := 8;
    C NUMBER := 3;
BEGIN
    IF A > B AND A > C THEN
        DBMS_OUTPUT.PUT_LINE('A is the greatest');
    ELSIF B > A AND B > C THEN
        DBMS_OUTPUT.PUT_LINE('B is the greatest');
    ELSE
        DBMS_OUTPUT.PUT_LINE('C is the greatest');
    END IF;
END;
/
```

```
B is the greatest
```

2. Using PL/SQL Statements create a simple loop that displays the message "Welcome to PL/SQL Programming" 20 times.

```
BEGIN
    FOR i IN 1..20 LOOP
        DBMS_OUTPUT.PUT_LINE('Welcome to PL/SQL Programming');
    END LOOP;
END;
/
```

```
Welcome to PL/SQL Programming
Welcome to PL/SQL Programming
Welcome to PL/SQL Programming
Welcome to PL/SQL Programming
Welcome to PL/SQL Programming
Welcome to PL/SQL Programming
Welcome to PL/SQL Programming
Welcome to PL/SQL Programming
Welcome to PL/SQL Programming
Welcome to PL/SQL Programming
Welcome to PL/SQL Programming
Welcome to PL/SQL Programming
Welcome to PL/SQL Programming
Welcome to PL/SQL Programming
Welcome to PL/SQL Programming
Welcome to PL/SQL Programming
Welcome to PL/SQL Programming
Welcome to PL/SQL Programming
Welcome to PL/SQL Programming
Welcome to PL/SQL Programming
```

3. PL/SQL Code to Find the Factorial of a Number

```
DECLARE
    num NUMBER := 5;  -- Input number
    fact NUMBER := 1;  -- Initialize factorial to 1
BEGIN
    FOR i IN 1..num LOOP
        fact := fact * i;
    END LOOP;
    DBMS_OUTPUT.PUT_LINE('Factorial of ' || num || ' is: ' || fact);
END;
/
```

```
Factorial of 5 is: 120
```

4. PL/SQL Program to Generate Fibonacci Series

```
DECLARE
    n NUMBER := 6;  -- Input value for the number of terms
    a NUMBER := 0;   -- First Fibonacci number
    b NUMBER := 1;   -- Second Fibonacci number
    c NUMBER;
BEGIN
    DBMS_OUTPUT.PUT_LINE('Fibonacci Series:');
    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;
/
```

```
Fibonacci Series:
0
1
1
2
3
5
```

5. PL/SQL Code to Find the Sum of First N Numbers

```
DECLARE
    N NUMBER := 5;  -- Input number
    sum NUMBER := 0; -- Initialize sum to 0
BEGIN
    FOR i IN 1..N LOOP
        sum := sum + i;
    END LOOP;
    DBMS_OUTPUT.PUT_LINE('Sum of first ' || N || ' numbers is: ' || sum);
END;
/
```

```
Sum of first 5 numbers is: 15
```