Assignment-8

PL/SQL - I:

1. Write a PL/SQL block of code to update salary of employee '7788' to 35000 if the salary is less than 35000.

```
SQL> DECLARE
  2      salary emp.sal%TYPE;
  3  BEGIN
  4      SELECT sal INTO salary
  5      FROM emp
  6      WHERE empno = 7788;
  7      IF salary < 35000 THEN
  8          UPDATE emp
  9          SET sal= 35000
 10          WHERE empno = 7788;
 11          DBMS_OUTPUT.PUT_LINE('Salary updated successfully.');
 12      ELSE
 13          DBMS_OUTPUT.PUT_LINE('Employee already has a salary of 35000 or more.');
 14      END IF;
 15  END;
 16  /

PL/SQL procedure successfully completed.
```

2. Write a PL/SQL block of code to insert all details of employee '7698' to a new table temp_emp, which has same structure as emp table.

```
SQL> CREATE TABLE temp_emp AS
  2  SELECT *
  3  FROM emp
  4  WHERE 1 = 2;

Table created.
```

3. Write a PL/SQL block of code to display ones name like "Hello <Name>', Whatever entered at run time.

```
SQL> DECLARE
  2      emp_details emp%ROWTYPE;
  3  BEGIN
  4      SELECT * INTO emp_details
  5      FROM emp
  6      WHERE empno = 7698;
  7      INSERT INTO temp_emp
  8      VALUES emp_details;
  9      DBMS_OUTPUT.PUT_LINE('Employee details inserted into temp_emp table successfully.');
 10  EXCEPTION
 11      WHEN OTHERS THEN
 12          DBMS_OUTPUT.PUT_LINE('Error: ' || SQLERRM);
 13  END;
 14  /

PL/SQL procedure successfully completed.
```

4. Write a PL/SQL block of code to print first 50 whole no.

```
SQL> DECLARE
  2      v_counter NUMBER := 1;
  3  BEGIN
  4      WHILE v_counter <= 50 LOOP
  5          DBMS_OUTPUT.PUT_LINE(v_counter);
  6          v_counter := v_counter + 1;
  7      END LOOP;
  8  END;
  9  /

PL/SQL procedure successfully completed.
```

5. Write a PL/SQL block of code to update the commission of employee number 7369 to Rs. 3000 if it NULL; else raise his commission by 25%.

```
SQL> DECLARE
  2      v_commission emp.comm%TYPE;
  3  BEGIN
  4      SELECT comm INTO v_commission
  5      FROM emp
  6      WHERE empno = 7369;
  7
  8      IF v_commission IS NULL THEN
  9          UPDATE emp
 10          SET comm = 3000
 11          WHERE empno = 7369;
 12
 13          DBMS_OUTPUT.PUT_LINE('Commission updated to Rs. 3000 for employee 7369.');
 14      ELSE
 15          UPDATE emp
 16          SET comm = comm * 1.25
 17          WHERE empno = 7369;
 18
 19          DBMS_OUTPUT.PUT_LINE('Commission increased by 25% for employee 7369.');
 20      END IF;
 21  END;
 22  /

PL/SQL procedure successfully completed.
```

6. Write a PL/SQL block of code to print even number between 1 and 10 using for loop.

```
SQL> DECLARE
  2       v_number NUMBER;
  3  BEGIN
  4       FOR v_number IN 1..10 LOOP
  5           IF MOD(v_number, 2) = 0 THEN
  6               DBMS_OUTPUT.PUT_LINE(v_number);
  7           END IF;
  8       END LOOP;
  9  END;
 10  /

PL/SQL procedure successfully completed.
```

7. Write a PL/SQL block of code that will allow 5% salary increment of an employee (emp number should be taken from user) if the employee working in the organization more than 22 year.

```
SQL> DECLARE
  2      v_empno emp.empno%TYPE;
  3      v_hiredate emp.hiredate%TYPE;
  4      v_years_worked NUMBER;
  5      v_increment_amount NUMBER;
  6  BEGIN
  7      v_empno := &emp_number;
  8
  9      SELECT hiredate INTO v_hiredate
 10      FROM emp
 11      WHERE empno = v_empno;
 12
 13      v_years_worked := TRUNC(MONTHS_BETWEEN(SYSDATE, v_hiredate) / 12);
 14
 15      IF v_years_worked > 22 THEN
 16          SELECT sal * 0.05 INTO v_increment_amount
 17          FROM emp
 18          WHERE empno = v_empno;
 19
 20          UPDATE emp
 21          SET sal = sal + v_increment_amount
 22          WHERE empno = v_empno;
 23
 24          DBMS_OUTPUT.PUT_LINE('Salary incremented by ' || v_increment_amount || ' for employee ' || v_empno);
 25      ELSE
 26          DBMS_OUTPUT.PUT_LINE('Employee has not worked for more than 22 years. No increment applied.');
 27      END IF;
 28  EXCEPTION
 29      WHEN NO_DATA_FOUND THEN
 30          DBMS_OUTPUT.PUT_LINE('Employee with empno ' || v_empno || ' not found.');
 31      WHEN OTHERS THEN
 32          DBMS_OUTPUT.PUT_LINE('An error occurred: ' || SQLERRM);
 33  END;
 34  /
Enter value for emp_number: 7369
old   7:      v_empno := &emp_number;
new   7:      v_empno := 7369;

PL/SQL procedure successfully completed.
```