University Institute of Engineering

Department of Computer Science & Engineering

Experiment: 3

Student Name: Habis Prakash Deka **UID:24BDA70031**

Branch: CSE **Section/Group: AIT-KRG-GP2**

Semester: 4th **Date of Performance:7/01/28**

Subject Name: DBMS

Aim of the practical:

To understand the basic structure of a PL/SQL program by creating and executing a simple PL/SQL block that includes declaration and execution sections, and to display output using built-in procedures.

Tool Used:

- **Database Management System:**
  Oracle Database

Objective:

To create a simple PL/SQL program demonstrating Declaration Section and Execution Section.

Practical / Experimental Steps

Step 1: Open Oracle SQL*Plus / SQL Developer and create a new SQL worksheet.

Step 2: Enable output display using the command: `SET SERVEROUTPUT ON`.

Step 3: Write the first PL/SQL block with a **DECLARE** section to define employee variables.

Step 4: Execute the block using **BEGIN…END** and display values using `DBMS_OUTPUT.PUT_LINE`.

Step 5: Run the second and third blocks to perform salary calculations and conditional tax bracket checks.

Step 6: Observe and verify the output results shown in the output window for each PL/SQL block execution

A. Declaration Section

```
DECLARE

   emp_id     NUMBER := 101;

   emp_name   VARCHAR2(50) := 'John Doe';

   emp_salary NUMBER := 90000;
```

B. Execution Section

```
BEGIN
   DBMS_OUTPUT.PUT_LINE('Employee ID : ' || emp_id);
   DBMS_OUTPUT.PUT_LINE('Employee Name : ' || emp_name);
   DBMS_OUTPUT.PUT_LINE('Employee Salary : ' || emp_salary);
END;
/
```

I / O Analysis

```
DECLARE
   emp_id     NUMBER := 101;
   emp_name   VARCHAR2(50) := 'John Doe';
   emp_salary NUMBER := 90000;
BEGIN
   DBMS_OUTPUT.PUT_LINE('Employee ID : ' || emp_id);
   DBMS_OUTPUT.PUT_LINE('Employee Name : ' || emp_name);
   DBMS_OUTPUT.PUT_LINE('Employee Salary : ' || emp_salary);
END;
/
```

Output: 105 ms

```
Employee ID : 101
Employee Name : John Doe
Employee Salary : 90000
```

```
DECLARE
  emp_salary NUMBER := 50000;
  result NUMBER;
BEGIN
  DBMS_OUTPUT.PUT_LINE('Mathemathical Operations on salary ');
  result := emp_salary * 0.01;
  DBMS_OUTPUT.PUT_LINE('10% of salary :'|| result);
  DBMS_OUTPUT.PUT_LINE('Salary After Promotion : ' || (emp_salary+result));
  DBMS_OUTPUT.PUT_LINE('Salary Ater Demotion : ' || (emp_salary-result));
  DBMS_OUTPUT.PUT_LINE('Salary per annum. :' || (emp_salary*12));
END;
/
```

Output:

```
Mathemathical Operations on salary
10% of salary :500
Salary After Promotion : 50500
Salary Ater Demotion : 49500
Salary per annum. :600000
```

```
DECLARE
emp_salary NUMBER:= 60000;
package NUMBER;
BEGIN
  DBMS_OUTPUT.PUT_LINE('Employee Tax Bracket (calculated based on salary only, real figures might differ) :');
  package := emp_salary*12;
  IF package<300000 then
      DBMS_OUTPUT.PUT_LINE('Poor');
    elsif package<800000 then
      DBMS_OUTPUT.PUT_LINE('Still Poor but hopeful');
    elsif package<1200000 then
```

```
    DBMS_OUTPUT.PUT_LINE('Just avoiding tax');
  elsif package>1200000 then
    IF package<1500000 then
      DBMS_OUTPUT.PUT_LINE('Just avoided tax-free life');
    else
      DBMS_OUTPUT.PUT_LINE('RICH');
    end if;
  end if;
END;
```

Output: 109 ms

```
Employee Tax Bracket (calculated based on salary only, real figures might
differ) :
Still Poor but hopeful
```

Learning outcomes (What I have learnt):

- Understood the basic structure of a PL/SQL block, including the **DECLARE** and **BEGIN…END** sections.
- Learned how to declare and initialize variables for storing data values.
- Gained knowledge of using `DBMS_OUTPUT.PUT_LINE` to display results during program execution.
- Practiced performing mathematical operations on variables within PL/SQL programs.
- Developed understanding of decision-making using **IF–ELSIF–ELSE** conditional statements.
- Acquired practical insight into how PL/SQL can be used for simple database-related computations and logic implementation.