

Name: Mimanshu Gahlaut

UID: 24BAI70038

Course: BE-CSE (AI&ML)

Subject: Database Management System

Experiment 3: PL/SQL program

1. Aim of the Session

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.

2. Software Requirements:

- **Database:** Oracle SQL

3. Objective of the Session

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

4. Practical / Experiment Steps

The work was carried out through the following activities:

1. **Program Structure Definition:** Designed a basic PL/SQL block consisting of declaration and execution sections to understand the program flow.
2. **Variable Declaration:** Declared required variables in the `DECLARE` section with appropriate data types.
3. **Logic Implementation:** Wrote executable statements inside the `BEGIN...END` block to perform operations using declared variables.
4. **Output Display:** Used built-in procedures such as `DBMS_OUTPUT.PUT_LINE` to display results on the screen.
5. **Execution and Verification:** Executed the PL/SQL block and verified correct output generation and successful program execution.

5. Procedure of the Practical

Execution was performed in the following order:

1. **Environment Initialization:** Opened the Oracle SQL environment (SQL*Plus / SQL Developer) and connected to the database server.
2. **Session Configuration:** Enabled output display using the `SET SERVEROUTPUT ON` command.
3. **Program Preparation:** Wrote a basic PL/SQL block with DECLARE, BEGIN, and END sections.
4. **Variable Setup:** Declared required variables with suitable data types in the declaration section.
5. **Logic Execution:** Implemented executable statements inside the BEGIN block.
6. **Output Handling:** Used `DBMS_OUTPUT.PUT_LINE` to display execution results.
7. **Program Execution:** Ran the PL/SQL block to execute the program.
8. **Result Verification:** Verified the displayed output to ensure correct program behavior.
9. **Documentation:** Saved the PL/SQL script and recorded the output for submission.

6. I/O Analysis (Input / Output Analysis)

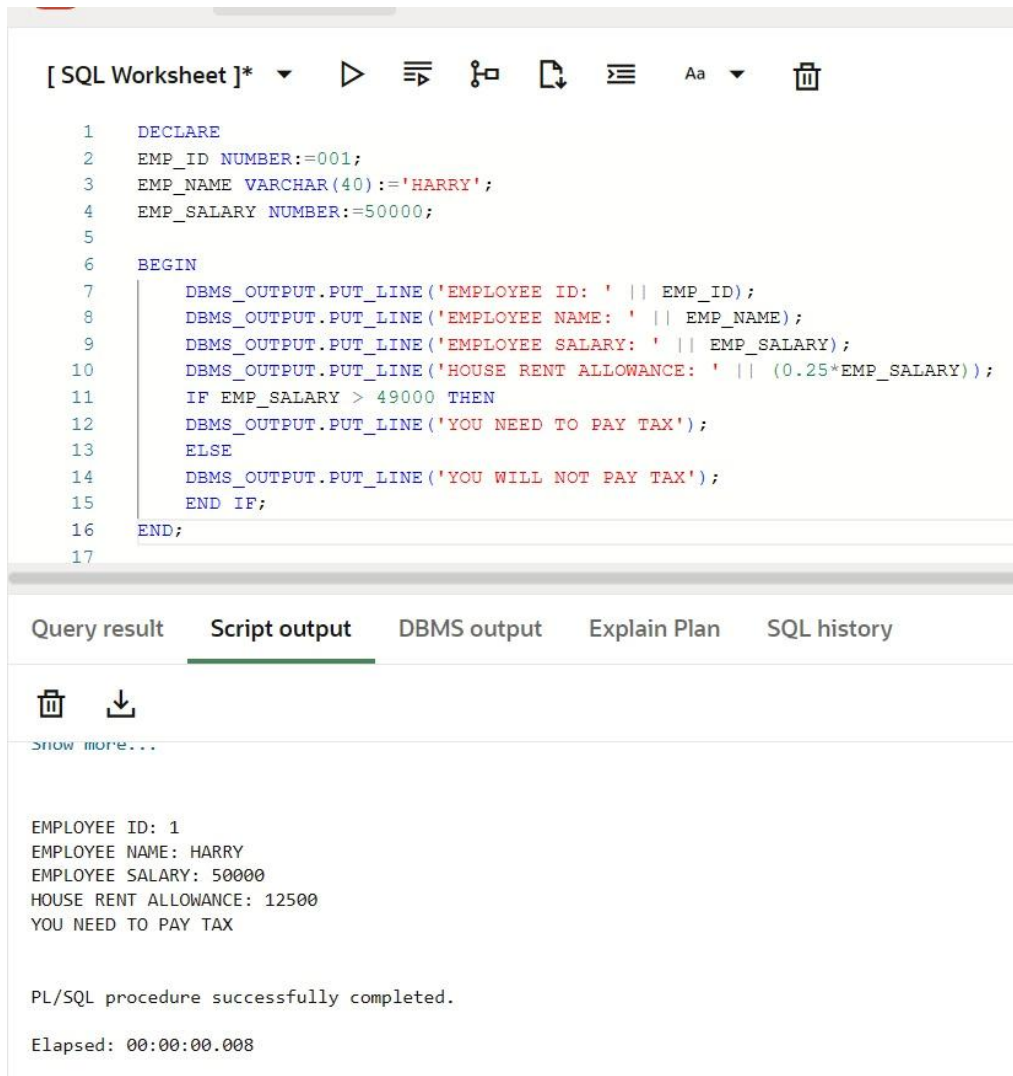
Input Queries

SQL

```
DECLARE
EMP_ID NUMBER:=001;
EMP_NAME VARCHAR(40):='HARRY';
EMP_SALARY NUMBER:=50000;

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);
    DBMS_OUTPUT.PUT_LINE('HOUSE RENT ALLOWANCE: ' || (0.25*EMP_SALARY));
    IF EMP_SALARY > 49000 THEN
        DBMS_OUTPUT.PUT_LINE('YOU NEED TO PAY TAX');
    ELSE
        DBMS_OUTPUT.PUT_LINE('YOU WILL NOT PAY TAX');
    END IF;
END;
```

Output Details



The screenshot displays an SQL worksheet interface. At the top, there is a toolbar with icons for running, saving, and other functions. Below the toolbar, the SQL code is entered in a text area, with line numbers 1 through 17 on the left. The code declares variables for EMP_ID, EMP_NAME, and EMP_SALARY, and then uses DBMS_OUTPUT.PUT_LINE to display their values. It also calculates a house rent allowance and checks if the employee needs to pay tax based on their salary. The output section at the bottom shows the results of the execution, including the employee details and the tax status. The interface also includes tabs for 'Query result', 'Script output', 'DBMS output', 'Explain Plan', and 'SQL history'.

```
1  DECLARE
2  EMP_ID NUMBER:=001;
3  EMP_NAME VARCHAR(40):='HARRY';
4  EMP_SALARY NUMBER:=50000;
5
6  BEGIN
7      DBMS_OUTPUT.PUT_LINE('EMPLOYEE ID: ' || EMP_ID);
8      DBMS_OUTPUT.PUT_LINE('EMPLOYEE NAME: ' || EMP_NAME);
9      DBMS_OUTPUT.PUT_LINE('EMPLOYEE SALARY: ' || EMP_SALARY);
10     DBMS_OUTPUT.PUT_LINE('HOUSE RENT ALLOWANCE: ' || (0.25*EMP_SALARY));
11     IF EMP_SALARY > 49000 THEN
12         DBMS_OUTPUT.PUT_LINE('YOU NEED TO PAY TAX');
13     ELSE
14         DBMS_OUTPUT.PUT_LINE('YOU WILL NOT PAY TAX');
15     END IF;
16 END;
```

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

EMPLOYEE ID: 1
EMPLOYEE NAME: HARRY
EMPLOYEE SALARY: 50000
HOUSE RENT ALLOWANCE: 12500
YOU NEED TO PAY TAX

PL/SQL procedure successfully completed.

Elapsed: 00:00:00.008

7. Learning Outcome

- Understood the basic structure of a PL/SQL program.
- Learned how to declare and use variables in PL/SQL.
- Gained experience in writing executable statements within a PL/SQL block.
- Used built-in procedures to display output.
- Developed basic procedural programming skills in PL/SQL.