

Name: Mimanshu Gahlaut

UID: 24BAI70038

Course: BE-CSE (AI&ML)

Subject: Database Management System

Experiment 4: PL/SQL conditional control statements

1. Aim of the Session

To design and implement PL/SQL programs utilizing conditional control statements such as IF–ELSE, ELSIF, ELSIF ladder, and CASE constructs in order to control the flow of execution based on logical conditions and to analyze decision-making capabilities in PL/SQL blocks.

2. Software Requirements:

- **Database:** Oracle SQL

3. Objective of the Session

Implement control structures in PL/SQL (IF-ELSE, ELSE-IF, ELSE-IF LADDER, CASE STATEMENTS in PL/SQL BLOCK).

4. Practical / Experiment Steps

The work was carried out through the following activities:

1. Open the Oracle SQL environment and connect to the database using valid credentials.
2. Create a new SQL worksheet for writing the PL/SQL program.
3. Declare required variables in the DECLARE section of the PL/SQL block.
4. Write the PL/SQL code using conditional control statements such as IF–ELSE, ELSIF ladder and CASE.
5. Observe and verify the output for different input values.

5. Procedure of the Practical

Execution was performed in the following order:

1. The Oracle SQL environment is opened and a connection to the database is established using valid user credentials.
2. Server output is enabled using the command SET SERVEROUTPUT ON to display the results of PL/SQL execution.
3. A PL/SQL block is written by declaring required variables in the DECLARE section.
4. Conditional control statements such as IF–ELSE, IF–ELSIF–ELSE, ELSIF ladder, and CASE are implemented in the BEGIN...END block according to the problem statements.
5. Output messages are displayed using the DBMS_OUTPUT.PUT_LINE statement based on evaluated conditions.
6. The PL/SQL program is executed and the output is observed.
7. The program is modified with different input values and re-executed to verify correctness.

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

Query 1: IF–ELSE (Check Positive/Non-Positive)

```
DECLARE
num NUMBER:= -3;

BEGIN
    IF num>0 THEN
        DBMS_OUTPUT.PUT_LINE('The number is positive');
    ELSE
        DBMS_OUTPUT.PUT_LINE('The number is not positive');
    END IF;
END;
```

Query 2: IF-ELSEIF-ELSE (Student Grading)

```
DECLARE
marks NUMBER:=78;

BEGIN
    IF marks>=90 THEN
        DBMS_OUTPUT.PUT_LINE('Grade: A');
    ELSIF marks>=75 THEN
        DBMS_OUTPUT.PUT_LINE('Grade: B');
    ELSIF marks>=60 THEN
        DBMS_OUTPUT.PUT_LINE('Grade: C');
    ELSE
        DBMS_OUTPUT.PUT_LINE('Grade: D');
    END IF;
END;
```

Query 3: ELSIF Ladder (Student Performance)

```
DECLARE
marks NUMBER:=85;

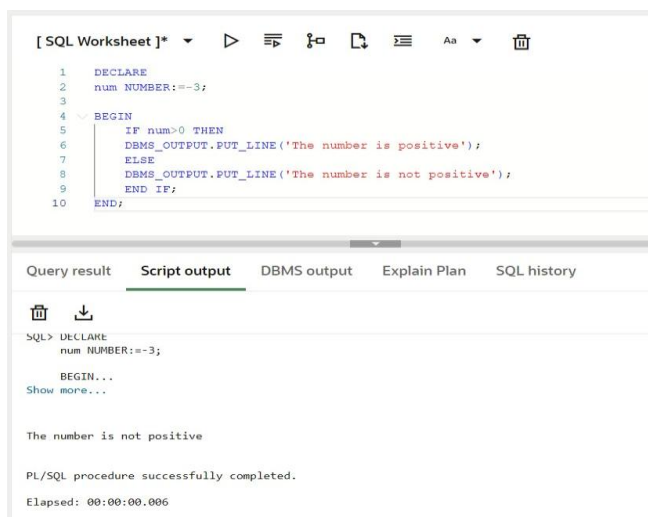
BEGIN
    IF marks>=90 THEN
        DBMS_OUTPUT.PUT_LINE('Excellent Performance');
    ELSIF marks>=75 THEN
        DBMS_OUTPUT.PUT_LINE('Very Good Performance');
    ELSIF marks>=60 THEN
        DBMS_OUTPUT.PUT_LINE('Good Performance');
    ELSIF marks>=40 THEN
        DBMS_OUTPUT.PUT_LINE('Average Performance');
    ELSE
        DBMS_OUTPUT.PUT_LINE('Poor Performance');
    END IF;
END;
```

Query 4: CASE Statement (Day of the Week)

```
DECLARE
num NUMBER:=3;

BEGIN
    CASE num
    WHEN 1 THEN DBMS_OUTPUT.PUT_LINE('Monday');
    WHEN 2 THEN DBMS_OUTPUT.PUT_LINE('Tuesday');
    WHEN 3 THEN DBMS_OUTPUT.PUT_LINE('Wednesday');
    WHEN 4 THEN DBMS_OUTPUT.PUT_LINE('Thursday');
    WHEN 5 THEN DBMS_OUTPUT.PUT_LINE('Friday');
    WHEN 6 THEN DBMS_OUTPUT.PUT_LINE('Saturday');
    WHEN 7 THEN DBMS_OUTPUT.PUT_LINE('Sunday');
    ELSE
        DBMS_OUTPUT.PUT_LINE('Invalid Day Number');
    END CASE;
END;
```

Output Details



The screenshot displays an SQL IDE interface. The top section shows a script editor with the following code:

```
1 DECLARE
2 num NUMBER:=3;
3
4 BEGIN
5     IF num>0 THEN
6         DBMS_OUTPUT.PUT_LINE('The number is positive');
7     ELSE
8         DBMS_OUTPUT.PUT_LINE('The number is not positive');
9     END IF;
10 END;
```

Below the script editor, the 'Script output' tab is active, showing the execution results:

```
SQL> DECLARE
      num NUMBER:=3;
      BEGIN...
      Show more...

      The number is not positive

      PL/SQL procedure successfully completed.
      Elapsed: 00:00:00.006
```

[SQL Worksheet]*

Aa

12 /

13

14 DECLARE

15 marks NUMBER:=78;

16

17 BEGIN

18 IF marks>=90 THEN

19 DBMS_OUTPUT.PUT_LINE('Grade: A');

20 ELSIF marks>=75 THEN

21 DBMS_OUTPUT.PUT_LINE('Grade: B');

22 ELSIF marks>=60 THEN

23 DBMS_OUTPUT.PUT_LINE('Grade: C');

24 ELSE

25 DBMS_OUTPUT.PUT_LINE('Grade: D');

26 END IF;

27 END;

Query result

Script output

DBMS output

Explain Plan

SQL history

SQL> DECLARE

marks NUMBER:=78;

BEGIN...

Show more...

Grade: B

PL/SQL procedure successfully completed.

Elapsed: 00:00:00.010

[SQL Worksheet]*

Aa

30

31 DECLARE

32 marks NUMBER:=85;

33

34 BEGIN

35 IF marks>=90 THEN

36 DBMS_OUTPUT.PUT_LINE('Excellent Performance');

37 ELSIF marks>=75 THEN

38 DBMS_OUTPUT.PUT_LINE('Very Good Performance');

39 ELSIF marks>=60 THEN

40 DBMS_OUTPUT.PUT_LINE('Good Performance');

41 ELSIF marks>=40 THEN

42 DBMS_OUTPUT.PUT_LINE('Average Performance');

43 ELSE

44 DBMS_OUTPUT.PUT_LINE('Poor Performance');

45 END IF;

46 END;

Query result

Script output

DBMS output

Explain Plan

SQL history

BEGIN...

Show more...

Very Good Performance

PL/SQL procedure successfully completed.

Elapsed: 00:00:00.008

[SQL Worksheet]*

Aa

48 /

49

50 DECLARE

51 num NUMBER:=3;

52

53 BEGIN

54 CASE num

55 WHEN 1 THEN DBMS_OUTPUT.PUT_LINE('Monday');

56 WHEN 2 THEN DBMS_OUTPUT.PUT_LINE('Tuesday');

57 WHEN 3 THEN DBMS_OUTPUT.PUT_LINE('Wednesday');

58 WHEN 4 THEN DBMS_OUTPUT.PUT_LINE('Thursday');

59 WHEN 5 THEN DBMS_OUTPUT.PUT_LINE('Friday');

60 WHEN 6 THEN DBMS_OUTPUT.PUT_LINE('Saturday');

61 WHEN 7 THEN DBMS_OUTPUT.PUT_LINE('Sunday');

62 ELSE

63 DBMS_OUTPUT.PUT_LINE('Invalid Day Number');

64 END CASE;

65 END;

Query result

Script output

DBMS output

Explain Plan

SQL history

BEGIN...

Show more...

Wednesday

PL/SQL procedure successfully completed.

Elapsed: 00:00:00.009

About Oracle

Contact Us

Legal Notices

Terms and Conditions

Your Privacy Rights

7. Learning Outcome

- Understand the concept of **conditional control statements** in PL/SQL.
- Implement **IF-ELSE** and **IF-ELSIF-ELSE** statements for decision making.
- Apply the **ELSIF ladder** to handle multiple conditions efficiently.
- Use the **CASE statement** for simplified and structured conditional logic.
- Analyze program flow based on logical conditions in PL/SQL blocks.