Student Name:DANISHMEET SINGH **UID: 24BAI70719**
Branch: AIT-CSE- AIML **Section/Group: 24AIT_KRG-1/G2**
Semester: 4 **Subject Name: DBMS**

AIM

To design and implement PL/SQL programs utilizing conditional control statements such as IF–ELSE, IF–ELSIF–ELSE, 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.

S/W Requirement:

- Database Management System: PostgreSQL / Oracle Database Express Edition
- Database Administration Tool: pgAdmin

OBJECTIVES:

- To understand and implement conditional control statements in PL/SQL
- To analyze decision-making using IF–ELSE, ELSIF ladder, and CASE statements
- To enhance logical thinking using PL/SQL blocks

PROBLEM STATEMENT:

Develop and execute PL/SQL programs that demonstrate the use of conditional control statements. The programs should employ IF–ELSE, IF–ELSIF–ELSE, ELSIF ladder, and CASE statements to evaluate given conditions and control the flow of execution accordingly.

1. PROBLEM STATEMENT – IF–ELSE STATEMENT

Write a PL/SQL program to check whether a given number is positive or non-positive using the IF–ELSE conditional control statement and display an appropriate message.

PROGRAM:

```
DECLARE
```

```
    num NUMBER := -5;
BEGIN
    IF num > 0 THEN
        DBMS_OUTPUT.PUT_LINE('The number is Positive');
    ELSE
        DBMS_OUTPUT.PUT_LINE('The number is Non-Positive');
    END IF;
END;
```

2. PROBLEM STATEMENT – IF–ELSIF–ELSE STATEMENT

Write a PL/SQL program to evaluate the grade of a student based on obtained marks and display the corresponding grade.

PROGRAM:

```
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: Fail');
    END IF;
END;
```

3. PROBLEM STATEMENT – ELSIF LADDER

Write a PL/SQL program to determine the performance status of a student based on marks using an ELSIF ladder.

PROGRAM:

```
DECLARE
   marks NUMBER := 82;
BEGIN
   IF marks >= 85 THEN
      DBMS_OUTPUT.PUT_LINE('Performance: Excellent');
   ELSIF marks >= 70 THEN
      DBMS_OUTPUT.PUT_LINE('Performance: Very Good');
   ELSIF marks >= 55 THEN
      DBMS_OUTPUT.PUT_LINE('Performance: Good');
   ELSIF marks >= 40 THEN
      DBMS_OUTPUT.PUT_LINE('Performance: Average');
   ELSE
      DBMS_OUTPUT.PUT_LINE('Performance: Poor');
   END IF;
END;
```

4. PROBLEM STATEMENT – CASE STATEMENT

Write a PL/SQL program to display the name of the day based on a given day number using the CASE statement.

PROGRAM:

```
DECLARE
   day_num NUMBER := 3;
   day_name VARCHAR2(20);
BEGIN
   CASE day_num
      WHEN 1 THEN day_name := 'Sunday';
      WHEN 2 THEN day_name := 'Monday';
      WHEN 3 THEN day_name := 'Tuesday';
      WHEN 4 THEN day_name := 'Wednesday';
```

```
    WHEN 5 THEN day_name := 'Thursday';
    WHEN 6 THEN day_name := 'Friday';
    WHEN 7 THEN day_name := 'Saturday';
    ELSE day_name := 'Invalid Day Number';
  END CASE;

  DBMS_OUTPUT.PUT_LINE('Day is: ' || day_name);
END;
```

LEARNING OUTCOMES:

1. Understood the use of conditional control statements in PL/SQL.
2. Learned to apply IF–ELSE and IF–ELSIF–ELSE statements for decision-making.
3. Implemented ELSIF ladder for evaluating multiple conditions.
4. Used CASE statements to simplify complex conditional logic.
5. Improved logical reasoning and procedural programming skills in PL/SQL.

OUTPUT :

```
1   DECLARE
2       num NUMBER := -5;
3   BEGIN
4       IF num > 0 THEN
5           DBMS_OUTPUT.PUT_LINE('The number is Positive');
6       ELSE
7           DBMS_OUTPUT.PUT_LINE('The number is Non-Positive');
8       END IF;
9   END;
10
```

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

```
SQL> DECLARE
         num NUMBER := -5;
     BEGIN
         IF num > 0 THEN...
Show more...


The number is Non-Positive


PL/SQL procedure successfully completed.

Elapsed: 00:00:00.006
```

```
12  DECLARE
13      marks NUMBER := 78;
14  BEGIN
15      IF marks >= 90 THEN
16          DBMS_OUTPUT.PUT_LINE('Grade: A');
17      ELSIF marks >= 75 THEN
18          DBMS_OUTPUT.PUT_LINE('Grade: B');
19      ELSIF marks >= 60 THEN
20          DBMS_OUTPUT.PUT_LINE('Grade: C');
21      ELSE
22          DBMS_OUTPUT.PUT_LINE('Grade: Fail');
23      END IF;
24  END;
25
26
```

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

```
SQL> DECLARE
         marks NUMBER := 78;
     BEGIN
         IF marks >= 90 THEN...
Show more...


Grade: B


PL/SQL procedure successfully completed.

Elapsed: 00:00:00.014
```


```
30  DECLARE
31      marks NUMBER := 82;
32  BEGIN
33      IF marks >= 85 THEN
34          DBMS_OUTPUT.PUT_LINE('Performance: Excellent');
35      ELSIF marks >= 70 THEN
36          DBMS_OUTPUT.PUT_LINE('Performance: Very Good');
37      ELSIF marks >= 55 THEN
38          DBMS_OUTPUT.PUT_LINE('Performance: Good');
39      ELSIF marks >= 40 THEN
40          DBMS_OUTPUT.PUT_LINE('Performance: Average');
41      ELSE
42          DBMS_OUTPUT.PUT_LINE('Performance: Poor');
43      END IF;
44  END;
45
46
```

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

```
SQL> DECLARE
         marks NUMBER := 82;
     BEGIN
         IF marks >= 85 THEN...
Show more...


Performance: Very Good


PL/SQL procedure successfully completed.

Elapsed: 00:00:00.008
```

```
48  DECLARE
49      day_num NUMBER := 3;
50      day_name VARCHAR2(20);
51  BEGIN
52      CASE day_num
53          WHEN 1 THEN day_name := 'Sunday';
54          WHEN 2 THEN day_name := 'Monday';
55          WHEN 3 THEN day_name := 'Tuesday';
56          WHEN 4 THEN day_name := 'Wednesday';
57          WHEN 5 THEN day_name := 'Thursday';
58          WHEN 6 THEN day_name := 'Friday';
59          WHEN 7 THEN day_name := 'Saturday';
60          ELSE day_name := 'Invalid Day Number';
61      END CASE;
62
63      DBMS_OUTPUT.PUT_LINE('Day is: ' || day_name);
64  END;
```

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

```
SQL> DECLARE
         day_num NUMBER := 3;
         day_name VARCHAR2(20);
     BEGIN...
Show more...


Day is: Tuesday


PL/SQL procedure successfully completed.

Elapsed: 00:00:00.013
```

CONCLUSION:

This experiment provided hands-on experience with conditional control statements in PL/SQL. The use of IF–ELSE, ELSIF ladder, and CASE statements helped in understanding decision-making mechanisms and control flow within PL/SQL programs.