**WEEK-2**

**PL/SQL programming**

**Exercise 1: Control Structures**

**Scenario 1:** Number Classification and Listing

**Question:**  
Write a PL/SQL block that accepts a number as input and performs the following tasks:

1. Check whether the number is positive, negative, or zero.
2. Check whether the number is even or odd.
3. If the number is positive, print all numbers from 1 to the entered number using a loop.

Use IF-ELSE, CASE, and FOR LOOP control structures to implement the logic.

**Code:**

SET SERVEROUTPUT ON;

DECLARE

num NUMBER := &Enter\_Number;

i NUMBER := 1;

BEGIN

IF num > 0 THEN

DBMS\_OUTPUT.PUT\_LINE('The number is positive.');

ELSIF num < 0 THEN

DBMS\_OUTPUT.PUT\_LINE('The number is negative.');

ELSE

DBMS\_OUTPUT.PUT\_LINE('The number is zero.');

END IF;

CASE MOD(num, 2)

WHEN 0 THEN

DBMS\_OUTPUT.PUT\_LINE('The number is even.');

WHEN 1 THEN

DBMS\_OUTPUT.PUT\_LINE('The number is odd.');

WHEN -1 THEN

DBMS\_OUTPUT.PUT\_LINE('The number is odd.');

END CASE;

IF num > 0 THEN

DBMS\_OUTPUT.PUT\_LINE('Numbers from 1 to ' || num || ':');

FOR i IN 1..num LOOP

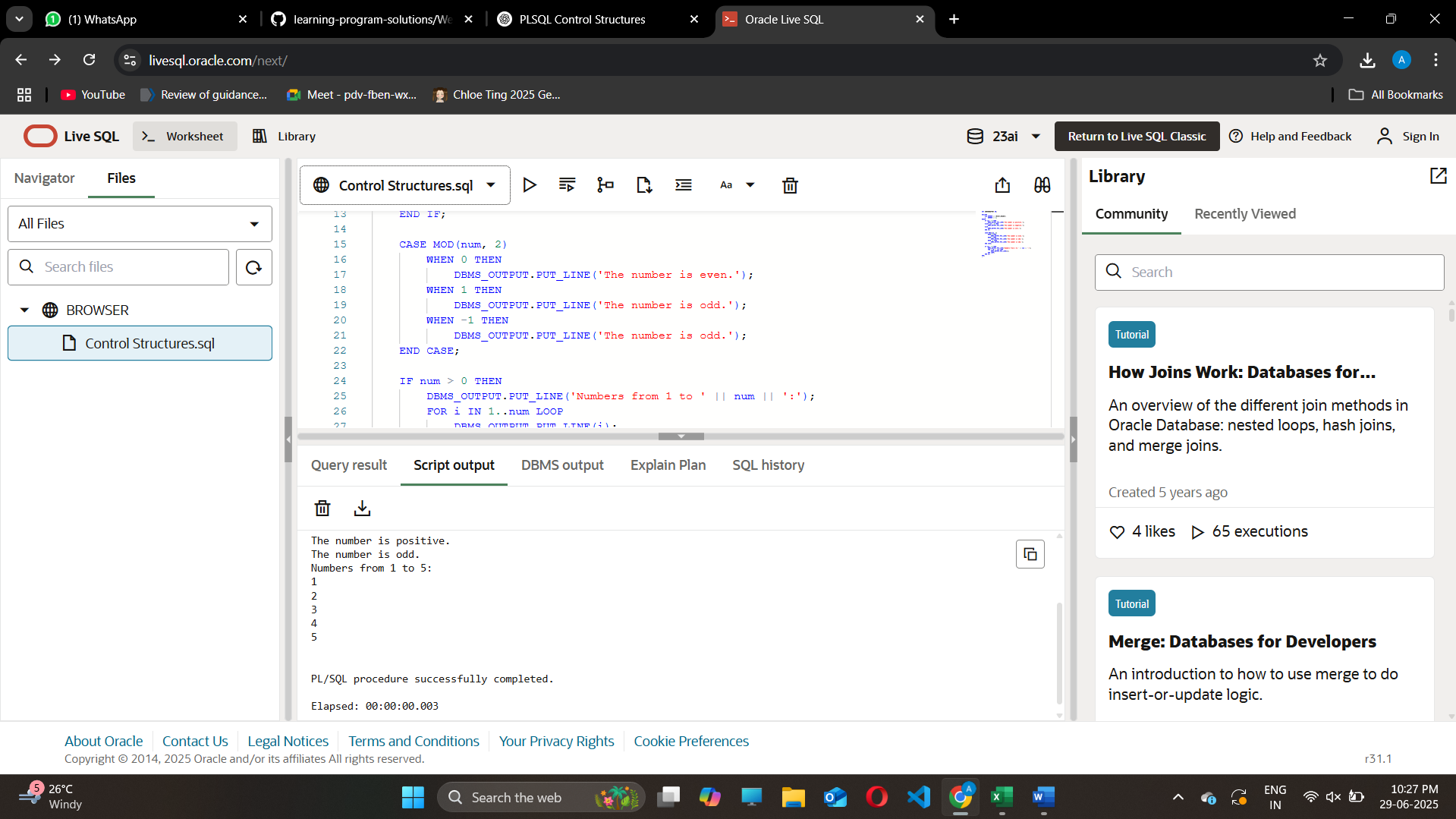
DBMS\_OUTPUT.PUT\_LINE(i);

END LOOP;

END IF;

END;

/

  
  
**Question**: Leap Year Check and Month Listing

**Scenario:** A school application needs to determine whether a given year is a leap year and then print all the months of the year.

1. Write a PL/SQL block that takes a year as input.
2. Check whether the year is a leap year using leap year conditions.
3. Print "Leap Year" or "Not a Leap Year" based on the result.
4. Use a loop and CASE statement to print all the months from January to December.

Use DBMS\_OUTPUT.PUT\_LINE for all outputs**.**

**Code:**

SET SERVEROUTPUT ON;

DECLARE

year\_input NUMBER := &Enter\_Year;

i NUMBER;

BEGIN

IF MOD(year\_input, 400) = 0 OR (MOD(year\_input, 4) = 0 AND MOD(year\_input, 100) != 0) THEN

DBMS\_OUTPUT.PUT\_LINE(year\_input || ' is a Leap Year.');

ELSE

DBMS\_OUTPUT.PUT\_LINE(year\_input || ' is NOT a Leap Year.');

END IF;

DBMS\_OUTPUT.PUT\_LINE('Months of the Year:');

FOR i IN 1..12 LOOP

CASE i

WHEN 1 THEN DBMS\_OUTPUT.PUT\_LINE('January');

WHEN 2 THEN DBMS\_OUTPUT.PUT\_LINE('February');

WHEN 3 THEN DBMS\_OUTPUT.PUT\_LINE('March');

WHEN 4 THEN DBMS\_OUTPUT.PUT\_LINE('April');

WHEN 5 THEN DBMS\_OUTPUT.PUT\_LINE('May');

WHEN 6 THEN DBMS\_OUTPUT.PUT\_LINE('June');

WHEN 7 THEN DBMS\_OUTPUT.PUT\_LINE('July');

WHEN 8 THEN DBMS\_OUTPUT.PUT\_LINE('August');

WHEN 9 THEN DBMS\_OUTPUT.PUT\_LINE('September');

WHEN 10 THEN DBMS\_OUTPUT.PUT\_LINE('October');

WHEN 11 THEN DBMS\_OUTPUT.PUT\_LINE('November');

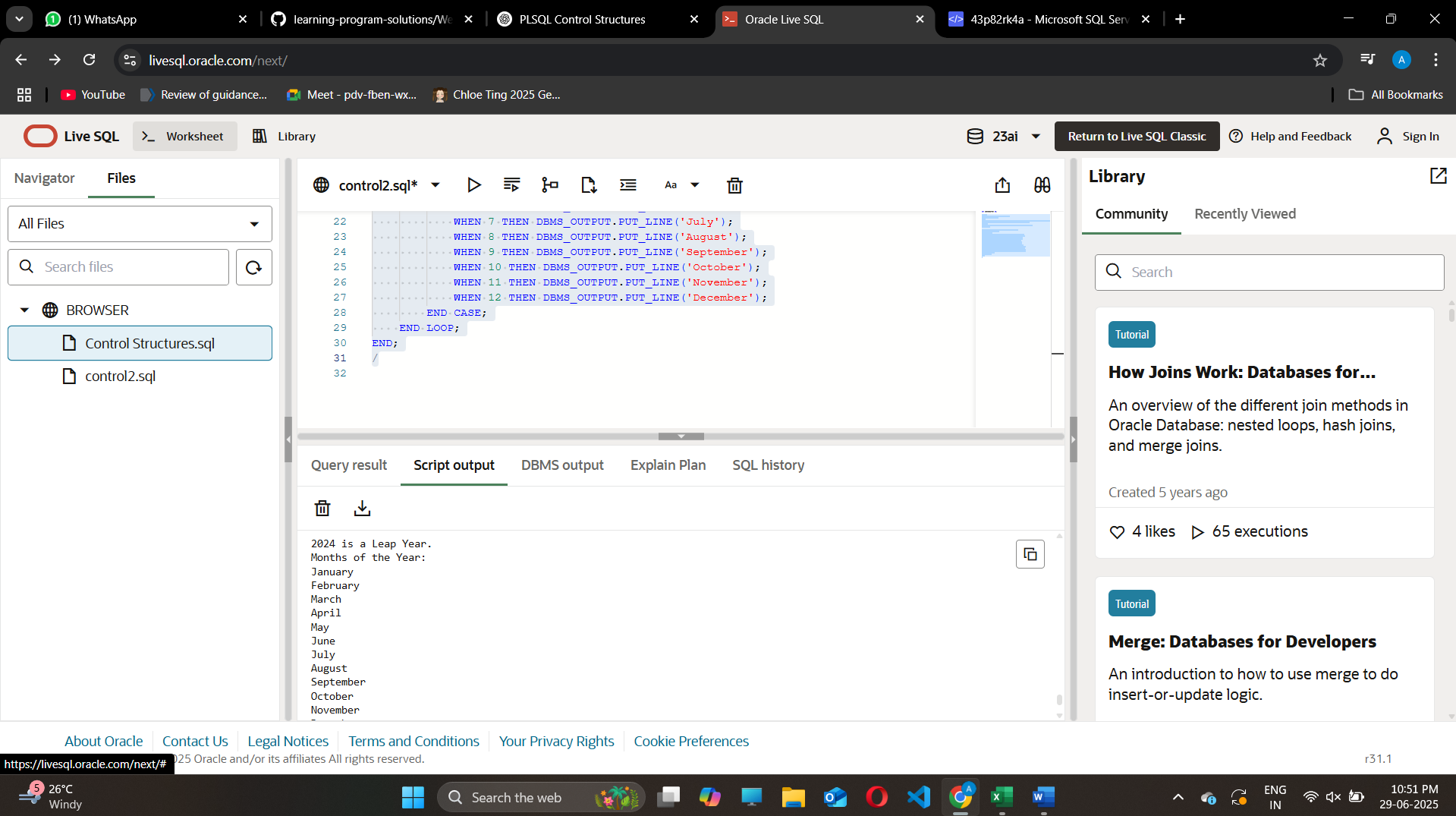
WHEN 12 THEN DBMS\_OUTPUT.PUT\_LINE('December');

END CASE;

END LOOP;

END;

/



**Exercise 3: Stored Procedures**

**Question:**

Write a PL/SQL program to create a procedure that displays the salary of an employee given their ID.

**Code:**

SET SERVEROUTPUT ON;

BEGIN

  EXECUTE IMMEDIATE 'DROP TABLE employees CASCADE CONSTRAINTS';

EXCEPTION

  WHEN OTHERS THEN

    NULL;

END;

/

CREATE TABLE employees (

  emp\_id NUMBER PRIMARY KEY,

  emp\_name VARCHAR2(50),

  salary NUMBER

);

/

BEGIN

  INSERT INTO employees VALUES (101, 'Amit', 50000);

  INSERT INTO employees VALUES (102, 'Nithya', 60000);

  INSERT INTO employees VALUES (103, 'Rahul', 70000);

  COMMIT;

END;

CREATE OR REPLACE PROCEDURE update\_salary (

  p\_emp\_id IN NUMBER,

  p\_percentage IN NUMBER

) AS

BEGIN

  UPDATE employees

  SET salary = salary + (salary \* p\_percentage / 100)

  WHERE emp\_id = p\_emp\_id;

  IF SQL%ROWCOUNT = 0 THEN

    DBMS\_OUTPUT.PUT\_LINE(' Employee ID ' || p\_emp\_id || ' not found.');

  ELSE

    DBMS\_OUTPUT.PUT\_LINE(' Salary updated for Employee ID: ' || p\_emp\_id);

  END IF;

EXCEPTION

  WHEN OTHERS THEN

    DBMS\_OUTPUT.PUT\_LINE('Error: ' || SQLERRM);

END;

/

BEGIN

  update\_salary(102, 10); -- Increase Nithya’s salary by 10%

END;

/

SELECT \* FROM employees;

**Output:**

A screenshot of a computer

AI-generated content may be incorrect.