6. Write a PL/SQL block of code using parameterized Cursor, that will merge the data available in the newly created table N_RollCall with the data available in the table O_RollCall. If the data in the first table already exists in the second table then that data should be skipped.

Solution:

Component	Purpose
`CURSOR`	Loops through `N_RollCall`
`IF NOT EXISTS`	Skips inserting if already exists
`student_id`	Used as the unique identifier

Step 1:

```
CREATE TABLE O_RollCall (
    student_id INT PRIMARY KEY,
    student_name VARCHAR(100),
    date_attended DATE
);

CREATE TABLE N_RollCall (
    student_id INT,
    student_name VARCHAR(100),
    date_attended DATE
);
```

Step 2:

```
DELIMITER //

CREATE PROCEDURE Merge_RollCall()
BEGIN
    DECLARE done INT DEFAULT FALSE;
    DECLARE v_id INT;
    DECLARE v_name VARCHAR(100);
    DECLARE v_date DATE;

    -- Cursor to loop through N_RollCall
    DECLARE cur CURSOR FOR
        SELECT student_id, student_name, date_attended FROM N_RollCall;

    DECLARE CONTINUE HANDLER FOR NOT FOUND SET done = TRUE;
    OPEN cur;

    read_loop: LOOP
        FETCH cur INTO v_id, v_name, v_date;

        IF done THEN
            LEAVE read_loop;
        END IF;

        -- Check if student_id exists in O_RollCall
        IF NOT EXISTS (SELECT 1 FROM O_RollCall WHERE student_id = v_id) THEN
            INSERT INTO O_RollCall (student_id, student_name, date_attended)
            VALUES (v_id, v_name, v_date);
        END IF;
        -- Else skip
    END LOOP;

    CLOSE cur;
END;
//

DELIMITER ;
```

Step 3:

```
CALL Merge_RollCall();

SELECT * FROM O_RollCall;
```