Program Code -

```
SET SERVEROUTPUT ON
BEGIN
DBMS_OUTPUT.PUT_LINE('========================');
  DBMS_OUTPUT.PUT_LINE('     CURSOR
MENU        ');
DBMS_OUTPUT.PUT_LINE('========================');
  DBMS_OUTPUT.PUT_LINE('1. Implicit
Cursor (1 row at a time)');
  DBMS_OUTPUT.PUT_LINE('2. Explicit
Cursor');
  DBMS_OUTPUT.PUT_LINE('3. Cursor FOR
Loop');
  DBMS_OUTPUT.PUT_LINE('4. Parameterized
Cursor');

DBMS_OUTPUT.PUT_LINE('========================');
END;
/


-- Accept user choice
ACCEPT choice NUMBER PROMPT 'Enter your
choice (1-4): '


-- PL/SQL block: reset O_RollCall, insert base
rows, merge
DECLARE
  v_choice NUMBER := &choice;

  CURSOR c_explicit IS
    SELECT RollNo, Name, Status FROM
N_RollCall;

  CURSOR c_param(p_roll NUMBER) IS
    SELECT RollNo, Name, Status FROM
N_RollCall WHERE RollNo = p_roll;

  v_rec N_RollCall%ROWTYPE;
BEGIN
  -- Reset O_RollCall
  EXECUTE IMMEDIATE 'TRUNCATE TABLE
O_RollCall';


  -- Insert base 3 rows
  INSERT INTO O_RollCall VALUES (1, 'Rahul',
'Present');
  INSERT INTO O_RollCall VALUES (2, 'Amit',
'Absent');
  INSERT INTO O_RollCall VALUES (3,
'Suresh', 'Present');
  COMMIT;


  -- Merge logic based on choice
  IF v_choice = 1 THEN
    -- Implicit cursor row-by-row: insert only 1
new row
    BEGIN
      DECLARE
        v_r N_RollCall%ROWTYPE;
      BEGIN
        SELECT RollNo, Name, Status
        INTO v_r.RollNo, v_r.Name, v_r.Status
        FROM N_RollCall n
        WHERE NOT EXISTS (SELECT 1
FROM O_RollCall o WHERE o.RollNo =
n.RollNo)
          AND ROWNUM = 1;

        INSERT INTO O_RollCall VALUES
(v_r.RollNo, v_r.Name, v_r.Status);
```

```sql
      EXCEPTION

      WHEN NO_DATA_FOUND THEN
NULL;

      END;

    END;


  ELSIF v_choice = 2 THEN

    -- Explicit cursor: loop through all rows

    OPEN c_explicit;

    LOOP

      FETCH c_explicit INTO v_rec.RollNo,
v_rec.Name, v_rec.Status;

      EXIT WHEN c_explicit%NOTFOUND;

      BEGIN

        INSERT INTO O_RollCall

        SELECT v_rec.RollNo, v_rec.Name,
v_rec.Status

        FROM dual

        WHERE NOT EXISTS (SELECT 1
FROM O_RollCall o WHERE o.RollNo =
v_rec.RollNo);

        EXCEPTION

        WHEN DUP_VAL_ON_INDEX THEN
NULL;

      END;

    END LOOP;

    CLOSE c_explicit;


  ELSIF v_choice = 3 THEN

    -- Cursor FOR loop

    FOR rec IN c_explicit LOOP

      BEGIN

        INSERT INTO O_RollCall

        SELECT rec.RollNo, rec.Name, rec.Status

        FROM dual

        WHERE NOT EXISTS (SELECT 1
FROM O_RollCall o WHERE o.RollNo =
rec.RollNo);

        EXCEPTION

        WHEN DUP_VAL_ON_INDEX THEN
NULL;

      END;

    END LOOP;


  ELSIF v_choice = 4 THEN

    -- Parameterized cursor

    FOR rec IN (SELECT RollNo FROM
N_RollCall) LOOP

      OPEN c_param(rec.RollNo);

      FETCH c_param INTO v_rec;

      IF v_rec.RollNo IS NOT NULL THEN

        BEGIN

          INSERT INTO O_RollCall

          SELECT v_rec.RollNo, v_rec.Name,
v_rec.Status

          FROM dual

          WHERE NOT EXISTS (SELECT 1
FROM O_RollCall o WHERE o.RollNo =
v_rec.RollNo);

          EXCEPTION

          WHEN DUP_VAL_ON_INDEX THEN
NULL;

        END;

      END IF;

      CLOSE c_param;

    END LOOP;

  END IF;

END;
/


SELECT RollNo, Name, Status FROM
O_RollCall ORDER BY RollNo;
```

Output –

```
SQL Plus                    ×    +    ∨

SQL> select * from O_RollCall;

    ROLLNO NAME                    STATUS
---------- -------------------- --------------------
         1 Rahul                   Present
         2 Amit                    Absent
         3 Suresh                  Present

SQL> select * from N_RollCall;

    ROLLNO NAME                    STATUS
---------- -------------------- --------------------
         2 Amit                    Absent
         3 Suresh                  Present
         4 Rohan                   Present
         5 Kiran                   Absent
         6 Aditya                  Present
         7 Arjun                   Absent

6 rows selected.

SQL> @C:\Users\kotka\Desktop\cursor.sql
=============================
CURSOR MENU
=============================
1. Implicit Cursor
2. Explicit Cursor
3. Cursor FOR Loop
4. Parameterized Cursor
```

```
SQL Plus                    ×    +    ∨

PL/SQL procedure successfully completed.

    ROLLNO NAME                    STATUS
---------- -------------------- --------------------
         1 Rahul                   Present
         2 Amit                    Absent
         3 Suresh                  Present
         4 Rohan                   Present

SQL> @C:\Users\kotka\Desktop\cursor.sql
=============================
CURSOR MENU
=============================
1. Implicit Cursor (1 row at a time)
2. Explicit Cursor
3. Cursor FOR Loop
4. Parameterized Cursor
=============================

PL/SQL procedure successfully completed.

Enter your choice (1-4): 4
old    2:      v_choice NUMBER := &choice;
new    2:      v_choice NUMBER :=          4;

PL/SQL procedure successfully completed.

    ROLLNO NAME                    STATUS
---------- -------------------- --------------------
         1 Rahul                   Present
         2 Amit                    Absent
         3 Suresh                  Present
         4 Rohan                   Present
         5 Kiran                   Absent
         6 Aditya                  Present
         7 Arjun                   Absent

7 rows selected.
```