ASSIGNMENT A7

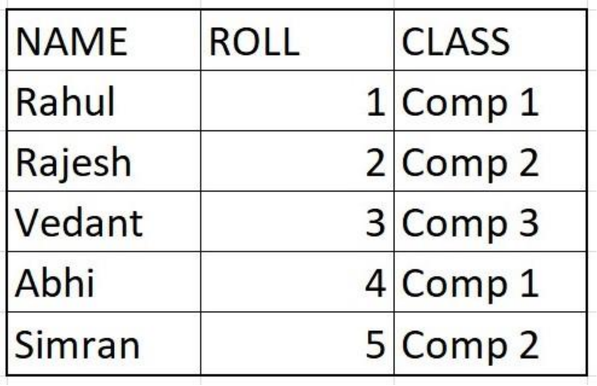
Cursors: (All types: Implicit, Explicit, Cursor FOR Loop, Parameterized Cursor)

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 exist in the second table then that data should be skipped.

```
-- Create tables and insert initial data
CREATE TABLE O_RollCall (
    name VARCHAR(255),
    roll NUMBER(14),
    class VARCHAR(255)
);

INSERT INTO O_RollCall VALUES ('Rahul', 1, 'Comp 1');
INSERT INTO O_RollCall VALUES ('Rajesh', 2, 'Comp 2');
INSERT INTO O_RollCall VALUES ('Vedant', 3, 'Comp 3');
INSERT INTO O_RollCall VALUES ('Abhi', 4, 'Comp 1');
INSERT INTO O_RollCall VALUES ('Simran', 5, 'Comp 2');

SELECT * FROM O_RollCall;

CREATE TABLE N_RollCall (
    name VARCHAR(255),
    roll NUMBER(14),
    class VARCHAR(255)
);

INSERT INTO N_RollCall VALUES ('Rahul', 1, 'Comp 1');
INSERT INTO N_RollCall VALUES ('Rajesh', 2, 'Comp 2');
INSERT INTO N_RollCall VALUES ('Vedant', 3, 'Comp 3');
INSERT INTO N_RollCall VALUES ('Abhi', 4, 'Comp 1');
INSERT INTO N_RollCall VALUES ('Simran', 5, 'Comp 2');
INSERT INTO N_RollCall VALUES ('Jay', 6, 'Comp 3');
INSERT INTO N_RollCall VALUES ('Geet', 7, 'Comp 2');

SELECT * FROM N_RollCall;
```

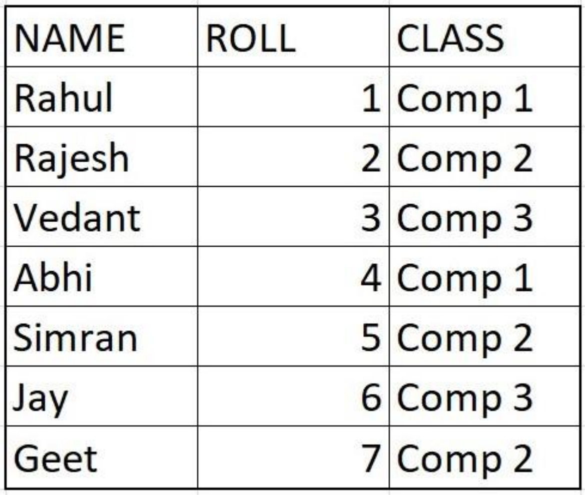
OUTPUT:

```
Table created.
1 row(s) inserted.
1 row(s) inserted.
1 row(s) inserted.
1 row(s) inserted.
1 row(s) inserted.
```



NAME	ROLL	CLASS
Rahul	1	Comp 1
Rajesh	2	Comp 2
Vedant	3	Comp 3
Abhi	4	Comp 1
Simran	5	Comp 2

```
5 rows selected.

Table created.
1 row(s) inserted.
1 row(s) inserted.
1 row(s) inserted.
1 row(s) inserted.
1 row(s) inserted.
1 row(s) inserted.
1 row(s) inserted.
```



NAME	ROLL	CLASS
Rahul	1	Comp 1
Rajesh	2	Comp 2
Vedant	3	Comp 3
Abhi	4	Comp 1
Simran	5	Comp 2
Jay	6	Comp 3
Geet	7	Comp 2

```
7 rows selected.
```

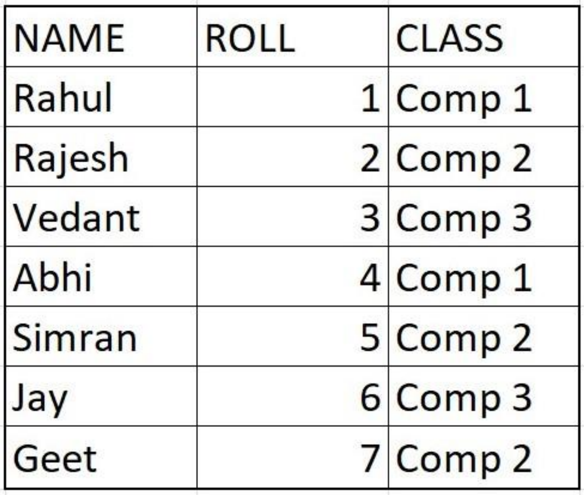
Using parameterized cursor.

```
-- PL/SQL block using parameterized cursor to merge data
DECLARE
    p_name VARCHAR(255);
    p_rollno NUMBER(15);
    p_class VARCHAR(255);

    CURSOR pp1(roll1 NUMBER) IS
        SELECT name, roll, class FROM N_RollCall
        WHERE roll > roll1
        AND roll NOT IN (SELECT roll FROM O_RollCall);
BEGIN
    -- Open cursor with parameter
    OPEN pp1(5); -- Adjust this value to set the roll number to start from

    LOOP
        FETCH pp1 INTO p_name, p_rollno, p_class;
        EXIT WHEN pp1%NOTFOUND;

        -- Insert only if not already present
        INSERT INTO O_RollCall VALUES (p_name, p_rollno, p_class);
        DBMS_OUTPUT.PUT_LINE('Inserted: ' || p_name || ' ' || p_rollno || ' ' ||
p_class);
    END LOOP;

    CLOSE pp1;
END;
/

-- Display contents of O_RollCall and N_RollCall after executing the PL/SQL
```

```
SELECT * FROM O_RollCall;
```

OUTPUT: (after running parameterized cursor.)

```
Statement processed.
Inserted: Jay 6 Comp 3
Inserted: Geet 7 Comp 2
```



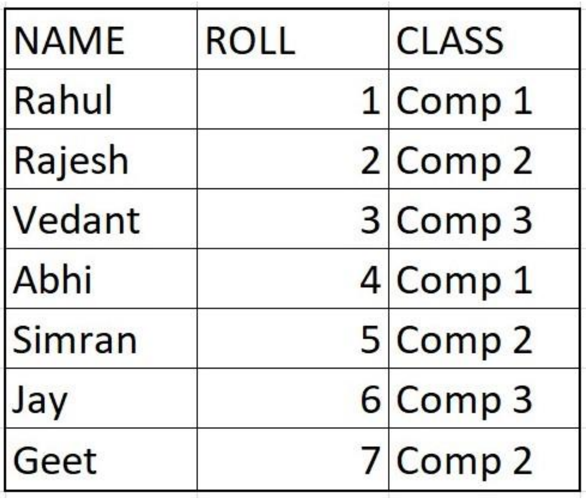
NAME	ROLL	CLASS
Rahul	1	Comp 1
Rajesh	2	Comp 2
Vedant	3	Comp 3
Abhi	4	Comp 1
Simran	5	Comp 2
Jay	6	Comp 3
Geet	7	Comp 2

Using explicit cursor.

```
-- PL/SQL block using explicit cursor to merge data
DECLARE
    p_name VARCHAR(255);
    p_rollno NUMBER(15);
    p_class VARCHAR(255);

    CURSOR cc1 IS
        SELECT * FROM N_RollCall
        WHERE roll NOT IN (SELECT roll FROM O_RollCall);
BEGIN
    OPEN cc1;
    LOOP
        FETCH cc1 INTO p_name, p_rollno, p_class;

        -- Exit the loop if no more rows
        EXIT WHEN cc1%NOTFOUND;

        -- Insert the record into O_RollCall
        INSERT INTO O_RollCall VALUES (p_name, p_rollno, p_class);
        DBMS_OUTPUT.PUT_LINE('Inserted: ' || p_name || ' ' || p_rollno || ' ' ||
p_class);
    END LOOP;

    CLOSE cc1;
END;
/
-- Display contents of O_RollCall and N_RollCall after executing the PL/SQL

SELECT * FROM O_RollCall;
```

OUTPUT: (after running explicit cursor.)

```
Statement processed.
Inserted: Jay 6 Comp 3
Inserted: Geet 7 Comp 2
```



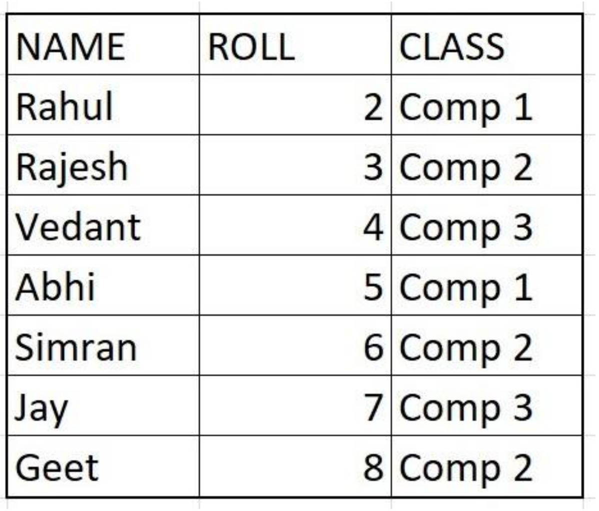
NAME	ROLL	CLASS
Rahul	1	Comp 1
Rajesh	2	Comp 2
Vedant	3	Comp 3
Abhi	4	Comp 1
Simran	5	Comp 2
Jay	6	Comp 3
Geet	7	Comp 2

Using implicit cursor.

```
-- PL/SQL block using implicit cursor to update roll numbers
DECLARE
    total_rows NUMBER(2);
BEGIN
    -- Update roll numbers in N_RollCall
    UPDATE N_RollCall SET roll = roll + 1;

    IF SQL%NOTFOUND THEN
        DBMS_OUTPUT.PUT_LINE('No roll was updated');
    ELSIF SQL%FOUND THEN
        total_rows := SQL%ROWCOUNT;
        DBMS_OUTPUT.PUT_LINE(total_rows || ' roll calls were affected');
    END IF;
END;
/
```

OUTPUT: (after running implicit cursor.)

```
Statement processed.
7 roll calls were affected
```



NAME	ROLL	CLASS
Rahul	2	Comp 1
Rajesh	3	Comp 2
Vedant	4	Comp 3
Abhi	5	Comp 1
Simran	6	Comp 2
Jay	7	Comp 3
Geet	8	Comp 2