**Group Project – DBAS32100**

**Part B – Task 7: PL/SQL Procedures**

**Student Name: DILPREET SINGH**

**Student ID: 991786155**

**Group: 03**

Selected Questions

I chose the following two questions from Part B, Task 7:

1. Question 2: What is the difference between DVD cost and the average cost of all DVDs of each year?
2. Question 3: Show the first name, movie name, and cumulative prices for each movie in increasing order.

I implemented both using PL/SQL procedures with analytic functions and cursors.

1. Procedure: DVD\_COST\_VS\_AVG\_BY\_YEAR

(Question 2)

This procedure shows:

* DVD title
* Year
* Cost
* Average cost of all DVDs released in the same year
* Difference between individual cost and average

HERE IS THE SQL CODE:

CREATE OR REPLACE PROCEDURE DVD\_COST\_VS\_AVG\_BY\_YEAR IS

CURSOR c1 IS

SELECT

Title,

Year,

Cost,

AVG(Cost) OVER (PARTITION BY Year) AS AvgYearCost,

Cost - AVG(Cost) OVER (PARTITION BY Year) AS Diff

FROM DVD

ORDER BY Year;

v\_rec c1%ROWTYPE;

BEGIN

DBMS\_OUTPUT.PUT\_LINE(RPAD('Title', 30) || ' | ' ||

RPAD('Year', 6) || ' | ' ||

RPAD('Cost', 8) || ' | ' ||

RPAD('Avg', 8) || ' | ' ||

'Diff');

DBMS\_OUTPUT.PUT\_LINE(RPAD('-', 70, '-'));

OPEN c1;

LOOP

FETCH c1 INTO v\_rec;

EXIT WHEN c1%NOTFOUND;

DBMS\_OUTPUT.PUT\_LINE(

RPAD(v\_rec.Title, 30) || ' | ' ||

RPAD(v\_rec.Year, 6) || ' | ' ||

RPAD(TO\_CHAR(v\_rec.Cost, '$999.99'), 8) || ' | ' ||

RPAD(TO\_CHAR(v\_rec.AvgYearCost, '$999.99'), 8) || ' | ' ||

TO\_CHAR(v\_rec.Diff, '$999.99')

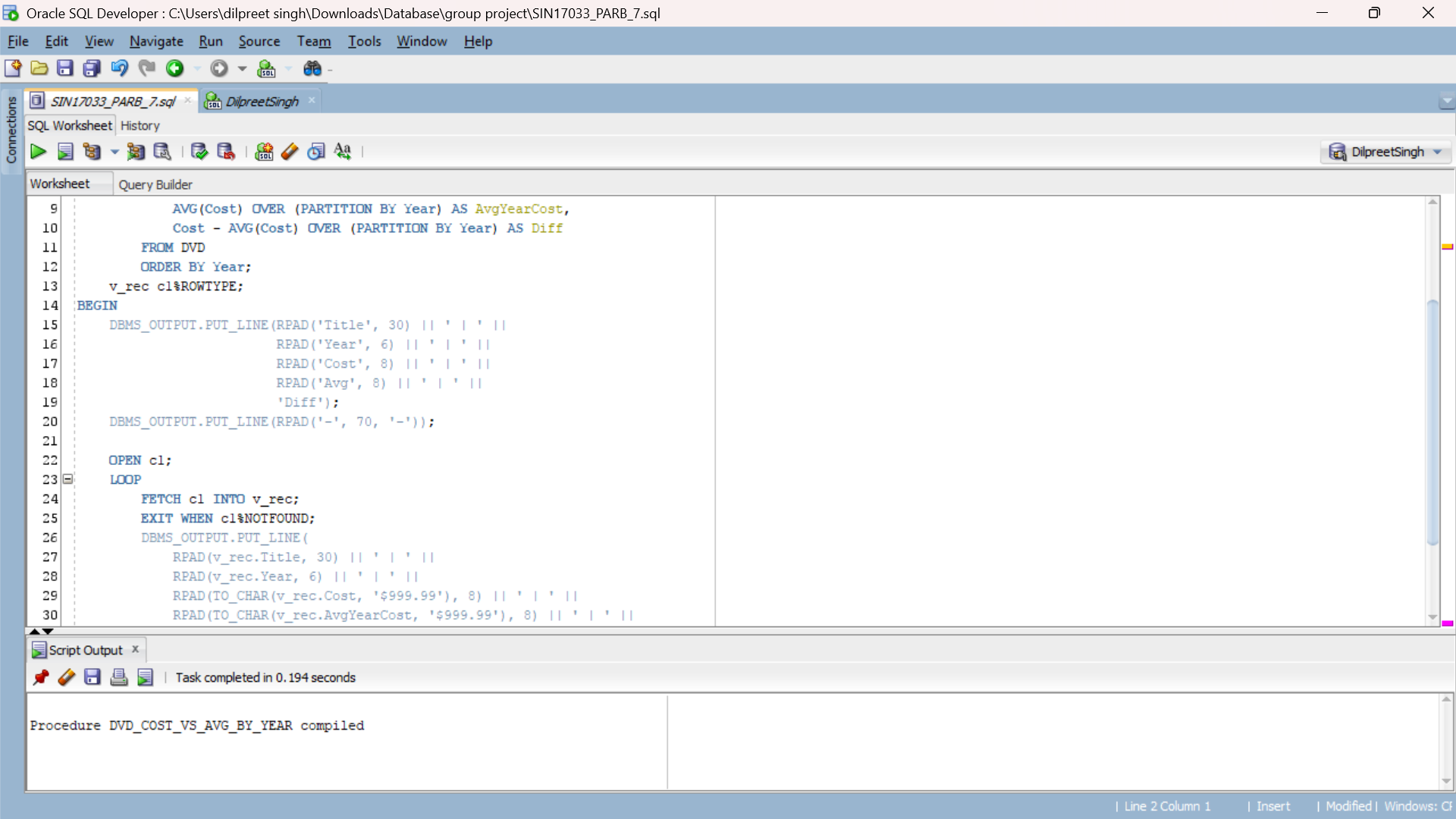
);

END LOOP;

CLOSE c1;

END;

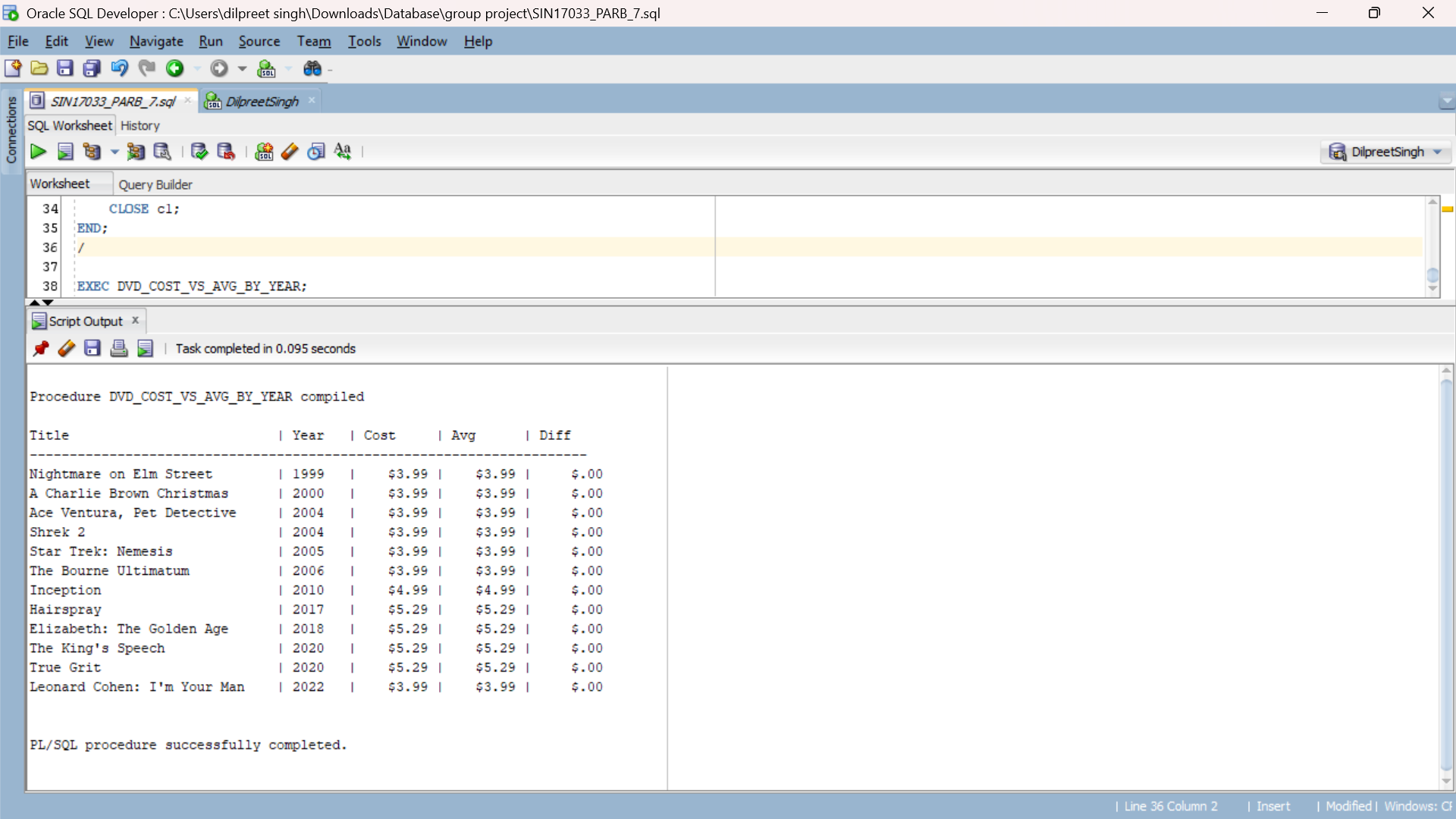
/



TEST SCRIPT:

SET SERVEROUTPUT ON;

EXEC DVD\_COST\_VS\_AVG\_BY\_YEAR;



2. Procedure: CUMULATIVE\_RENTAL\_BY\_MOVIE

(Question 3)

This procedure shows:

* Customer first name
* Movie title
* Rental cost
* Cumulative total of rental costs ordered by movie name

HERE IS THE SQL CODE:

CREATE OR REPLACE PROCEDURE CUMULATIVE\_RENTAL\_BY\_MOVIE IS

CURSOR c2 IS

SELECT

c.FirstName,

d.Title,

d.Cost,

SUM(d.Cost) OVER (ORDER BY d.Title, c.FirstName ROWS UNBOUNDED PRECEDING) AS RunningTotal

FROM DVD\_RENTAL r

JOIN CUSTOMER c ON r.CustomerID = c.CustomerID

JOIN DVD d ON r.TapeID = d.TapeID

ORDER BY d.Title, c.FirstName;

v\_row c2%ROWTYPE;

BEGIN

DBMS\_OUTPUT.PUT\_LINE(RPAD('Name', 15) || ' | ' ||

RPAD('Movie', 30) || ' | ' ||

RPAD('Cost', 8) || ' | ' ||

'Cumulative');

DBMS\_OUTPUT.PUT\_LINE(RPAD('-', 70, '-'));

OPEN c2;

LOOP

FETCH c2 INTO v\_row;

EXIT WHEN c2%NOTFOUND;

DBMS\_OUTPUT.PUT\_LINE(

RPAD(v\_row.FirstName, 15) || ' | ' ||

RPAD(v\_row.Title, 30) || ' | ' ||

RPAD(TO\_CHAR(v\_row.Cost, '$999.99'), 8) || ' | ' ||

TO\_CHAR(v\_row.RunningTotal, '$999.99')

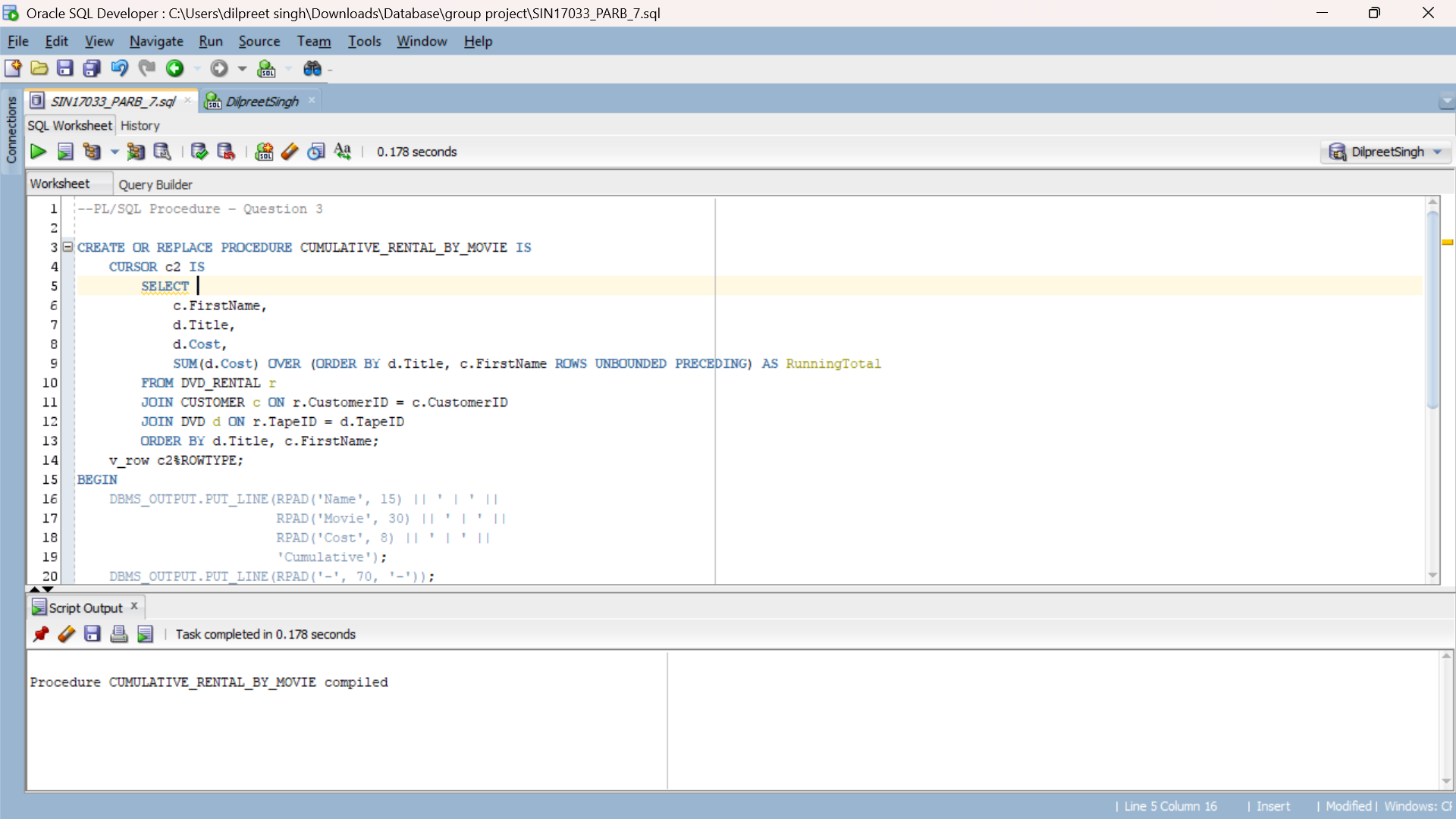
);

END LOOP;

CLOSE c2;

END;

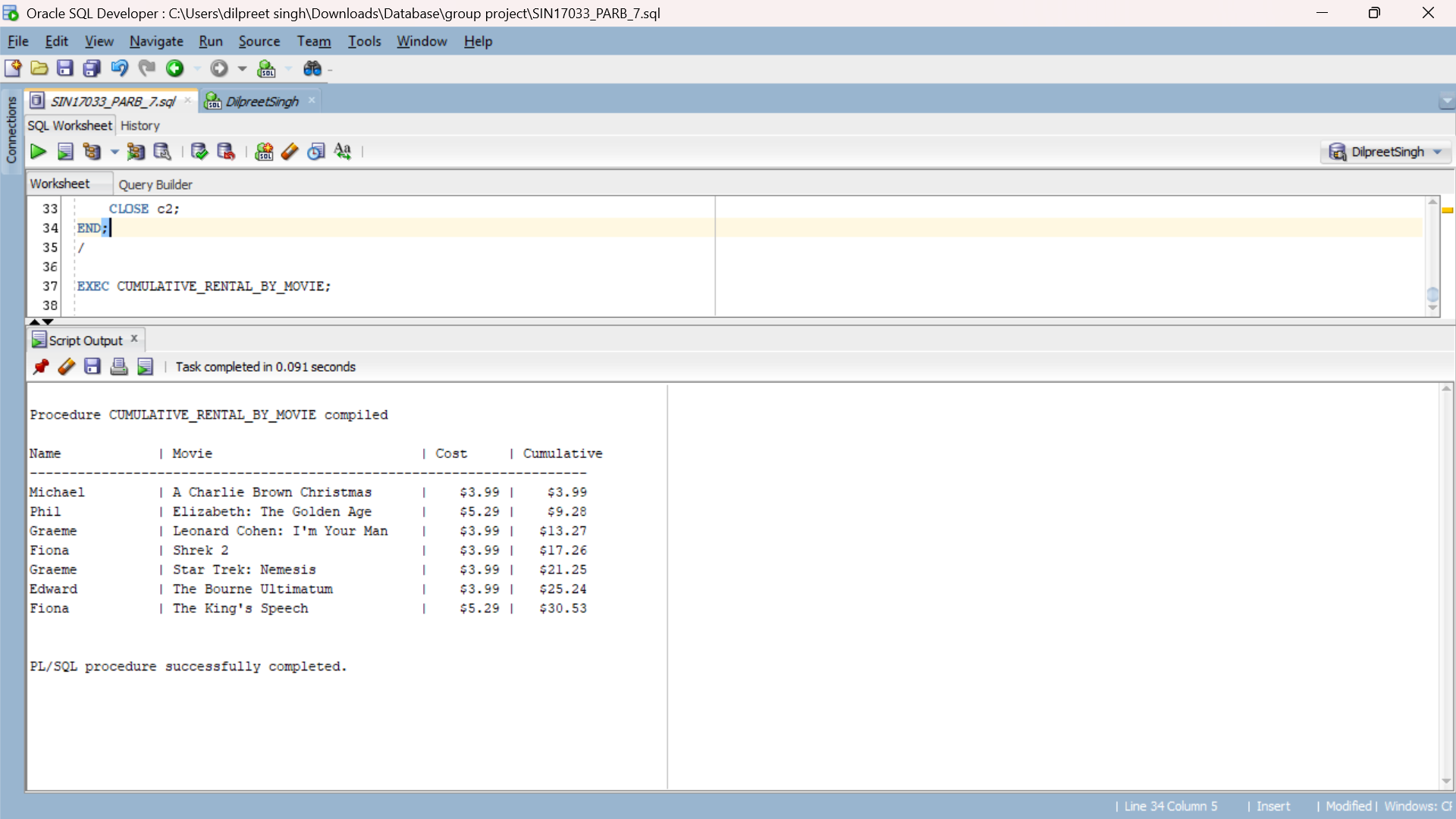
/



TEST SCRIPT:

SET SERVEROUTPUT ON;

EXEC CUMULATIVE\_RENTAL\_BY\_MOVIE;



Summary

* Both procedures use analytic functions (AVG() OVER, SUM() OVER) for calculations.
* Output is formatted using RPAD and TO\_CHAR for clear alignment.
* Tested with real data from the Excel file.
* Shared with group for integration into final SQL file and report.