**DBI LAB 4**

FULL NAME: DƯƠNG HỒNG QUÂN

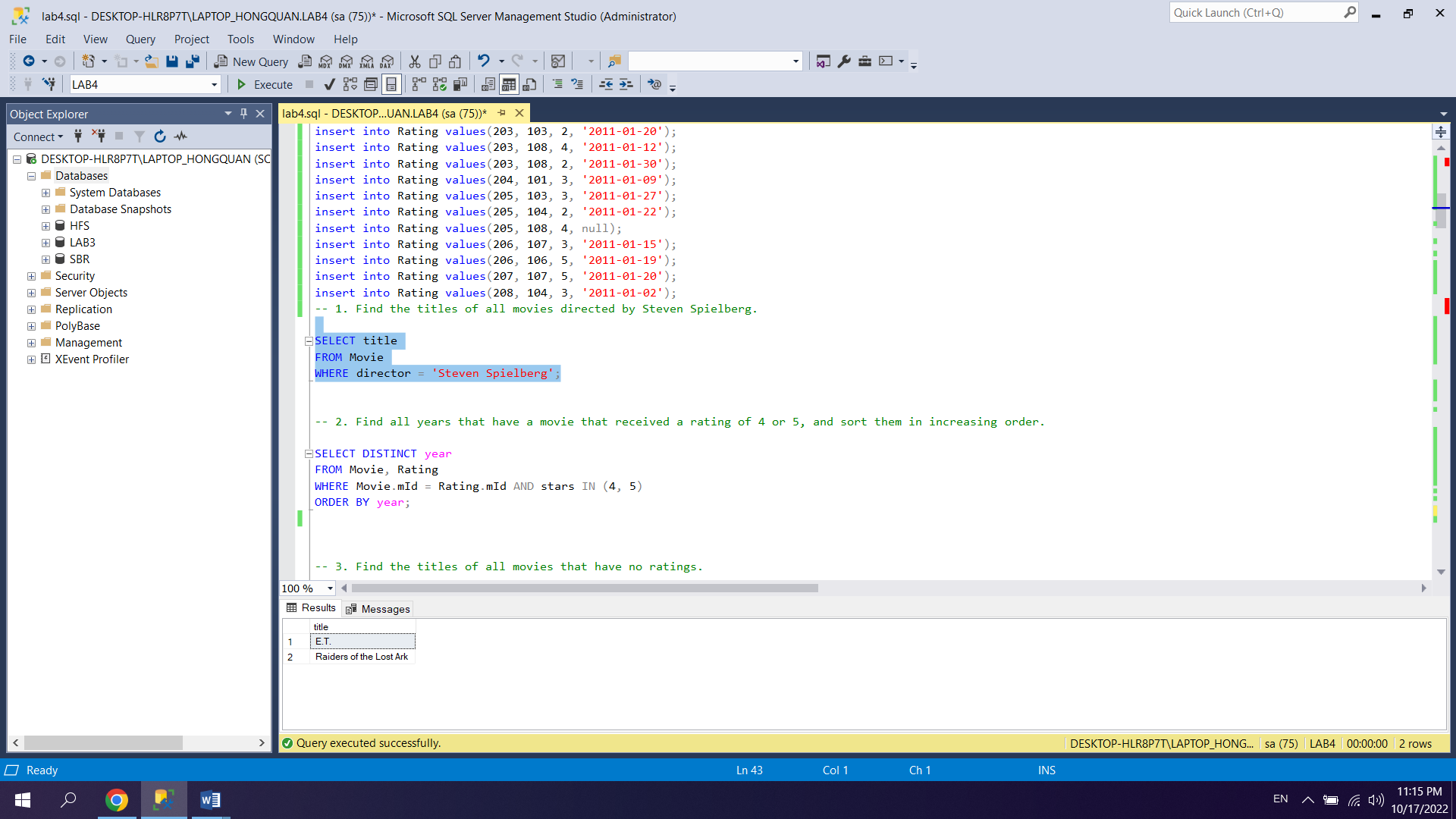
STUDENT ID: SE170057

1.

SELECT title

FROM Movie

WHERE director = 'Steven Spielberg';



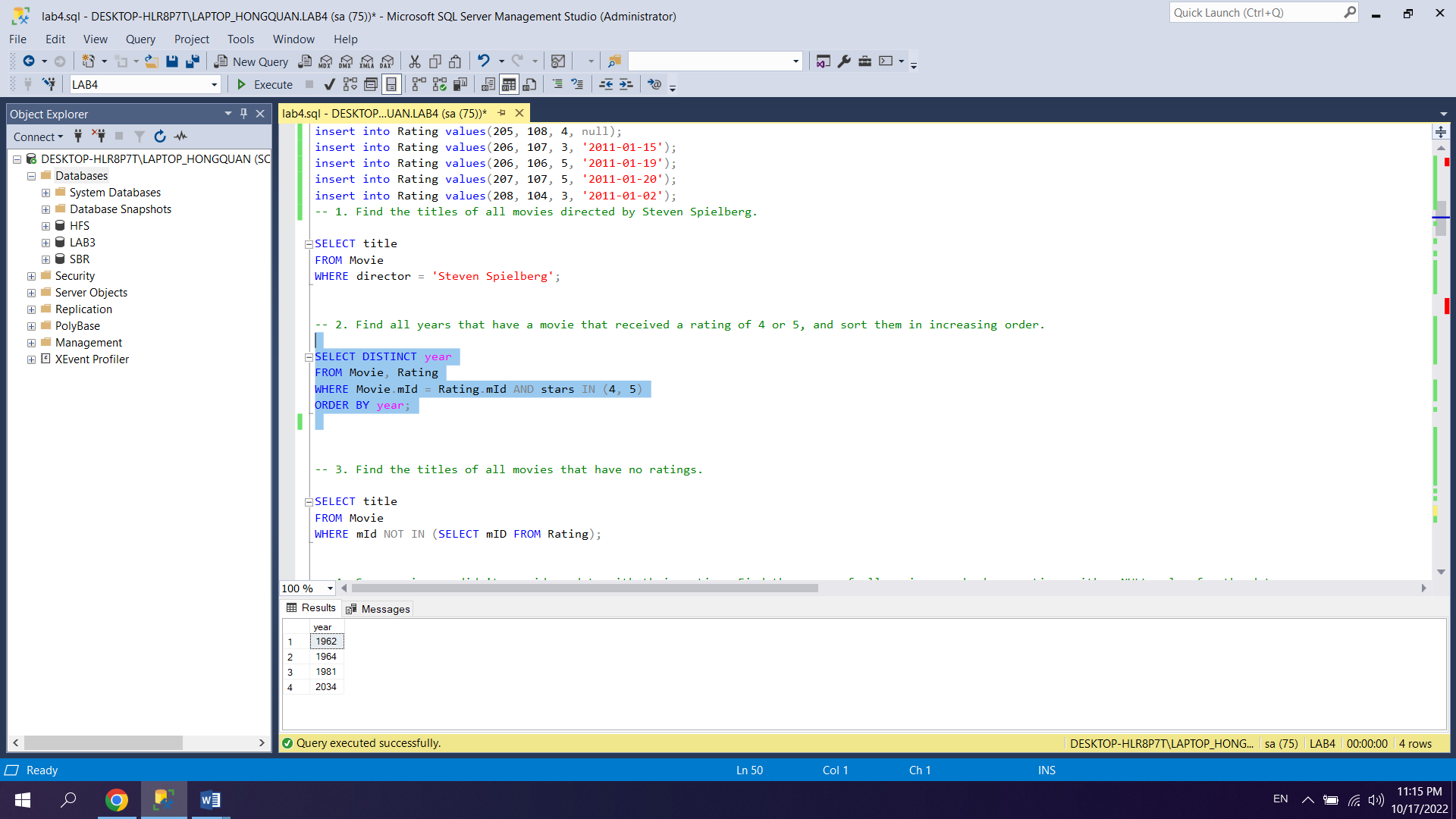
2.

SELECT DISTINCT year

FROM Movie, Rating

WHERE Movie.mId = Rating.mId AND stars IN (4, 5)

ORDER BY year;

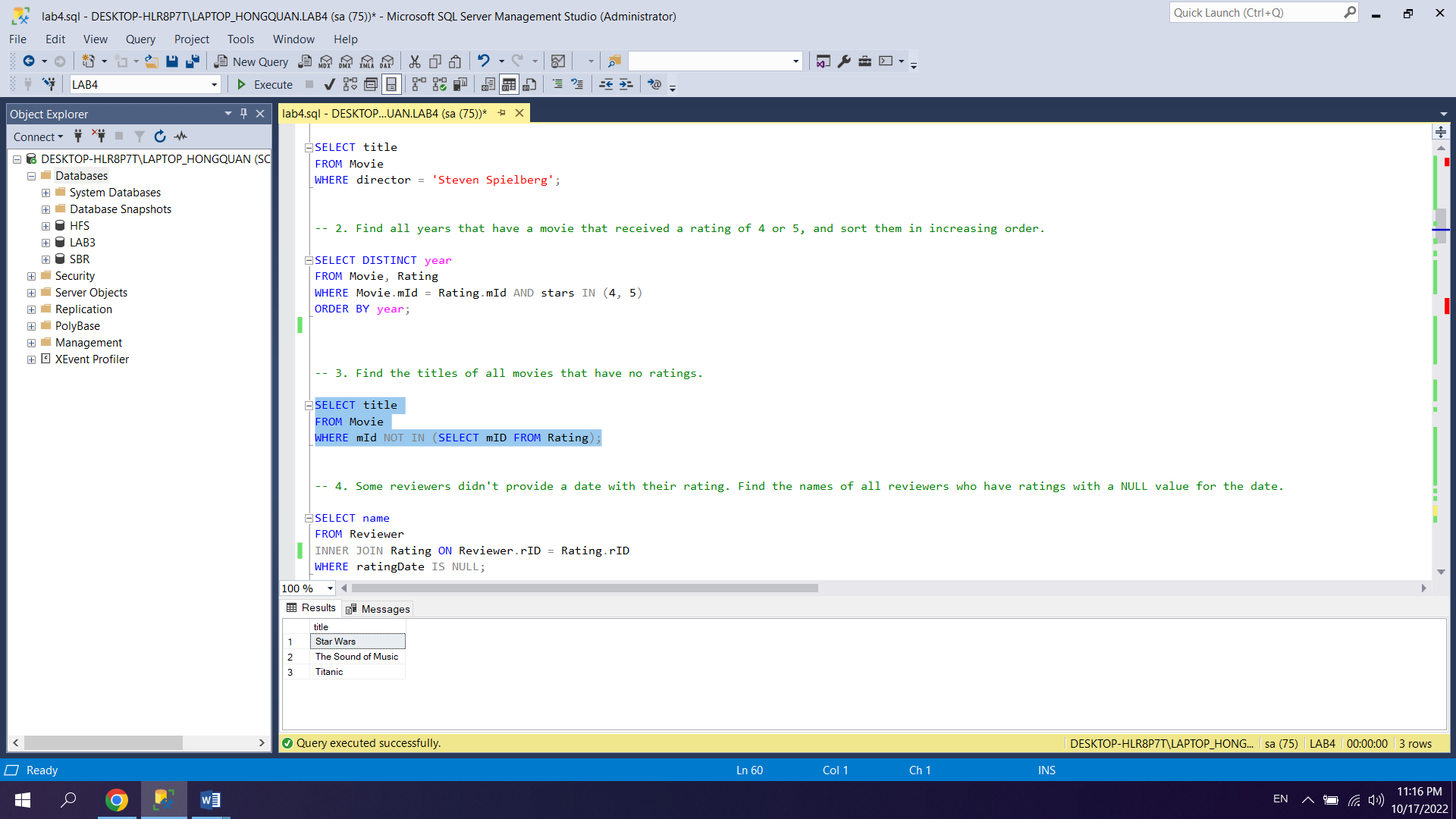


3.

SELECT title

FROM Movie

WHERE mId NOT IN (SELECT mID FROM Rating);



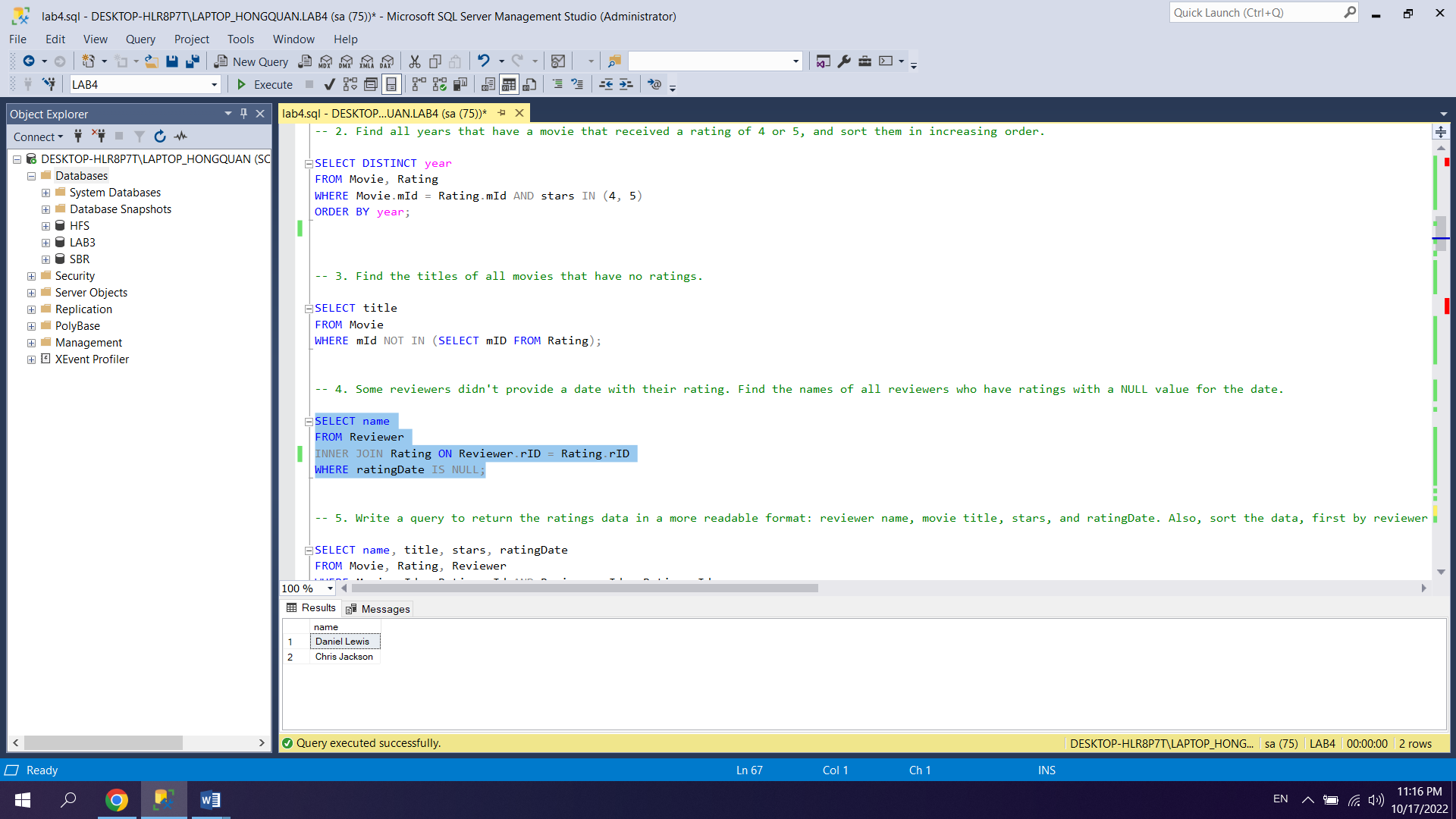
4.

SELECT name

FROM Reviewer

INNER JOIN Rating ON Reviewer.rID = Rating.rID

WHERE ratingDate IS NULL;



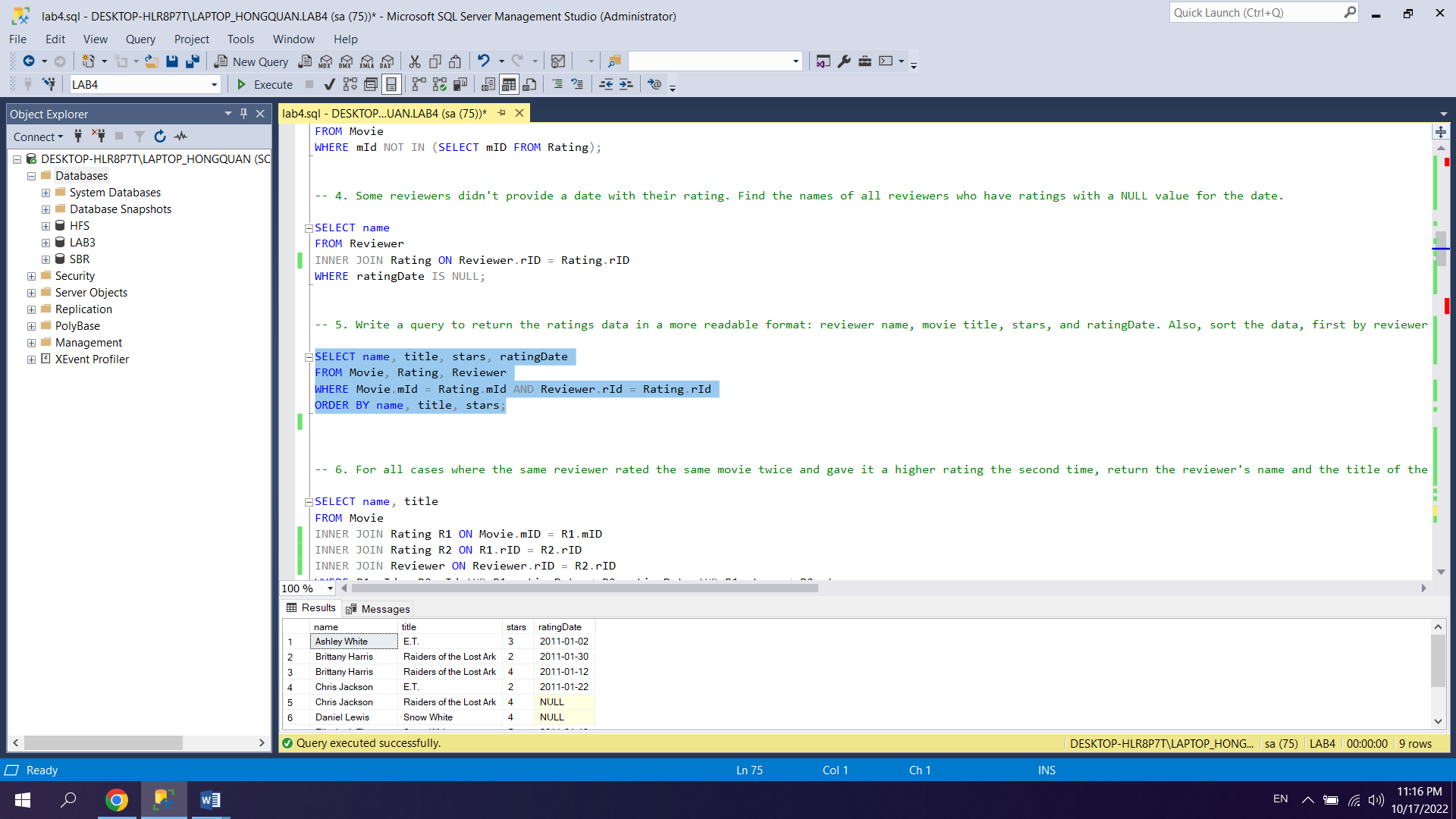
5.

SELECT name, title, stars, ratingDate

FROM Movie, Rating, Reviewer

WHERE Movie.mId = Rating.mId AND Reviewer.rId = Rating.rId

ORDER BY name, title, stars;



6.

SELECT name, title

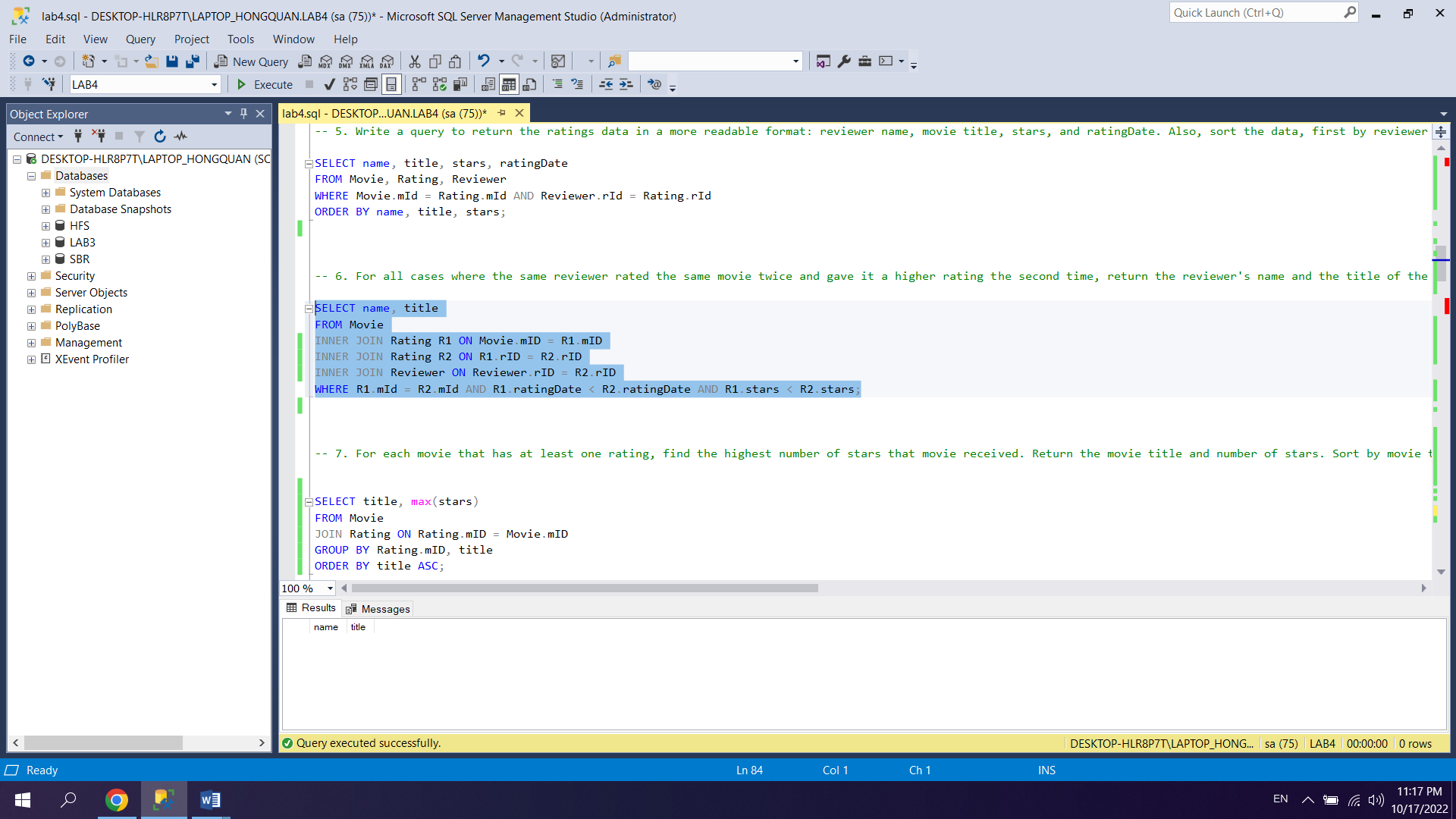
FROM Movie

INNER JOIN Rating R1 ON Movie.mID = R1.mID

INNER JOIN Rating R2 ON R1.rID = R2.rID

INNER JOIN Reviewer ON Reviewer.rID = R2.rID

WHERE R1.mId = R2.mId AND R1.ratingDate < R2.ratingDate AND R1.stars < R2.stars;



7.

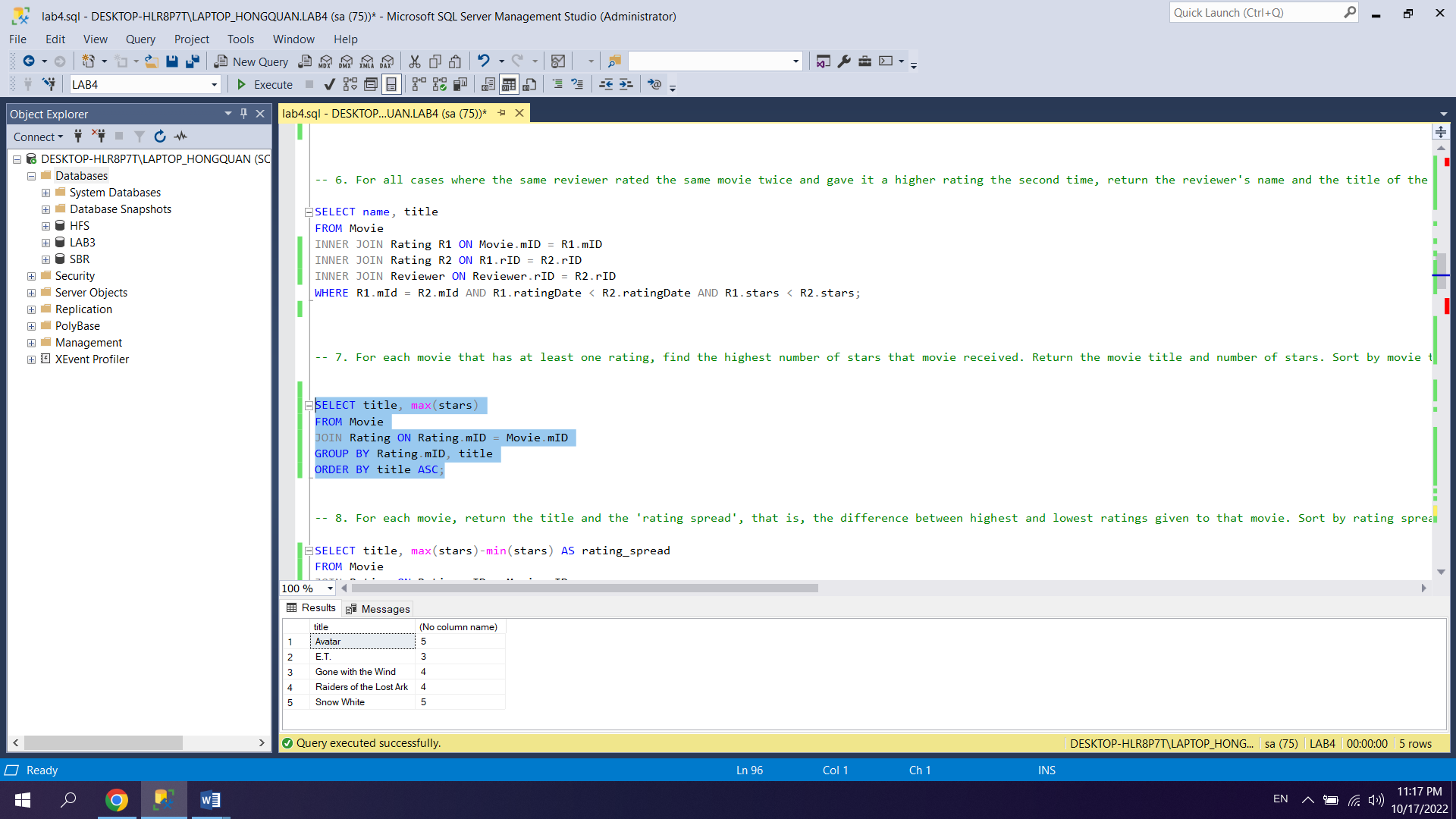
SELECT title, max(stars)

FROM Movie

JOIN Rating ON Rating.mID = Movie.mID

GROUP BY Rating.mID, title

ORDER BY title ASC;



8.

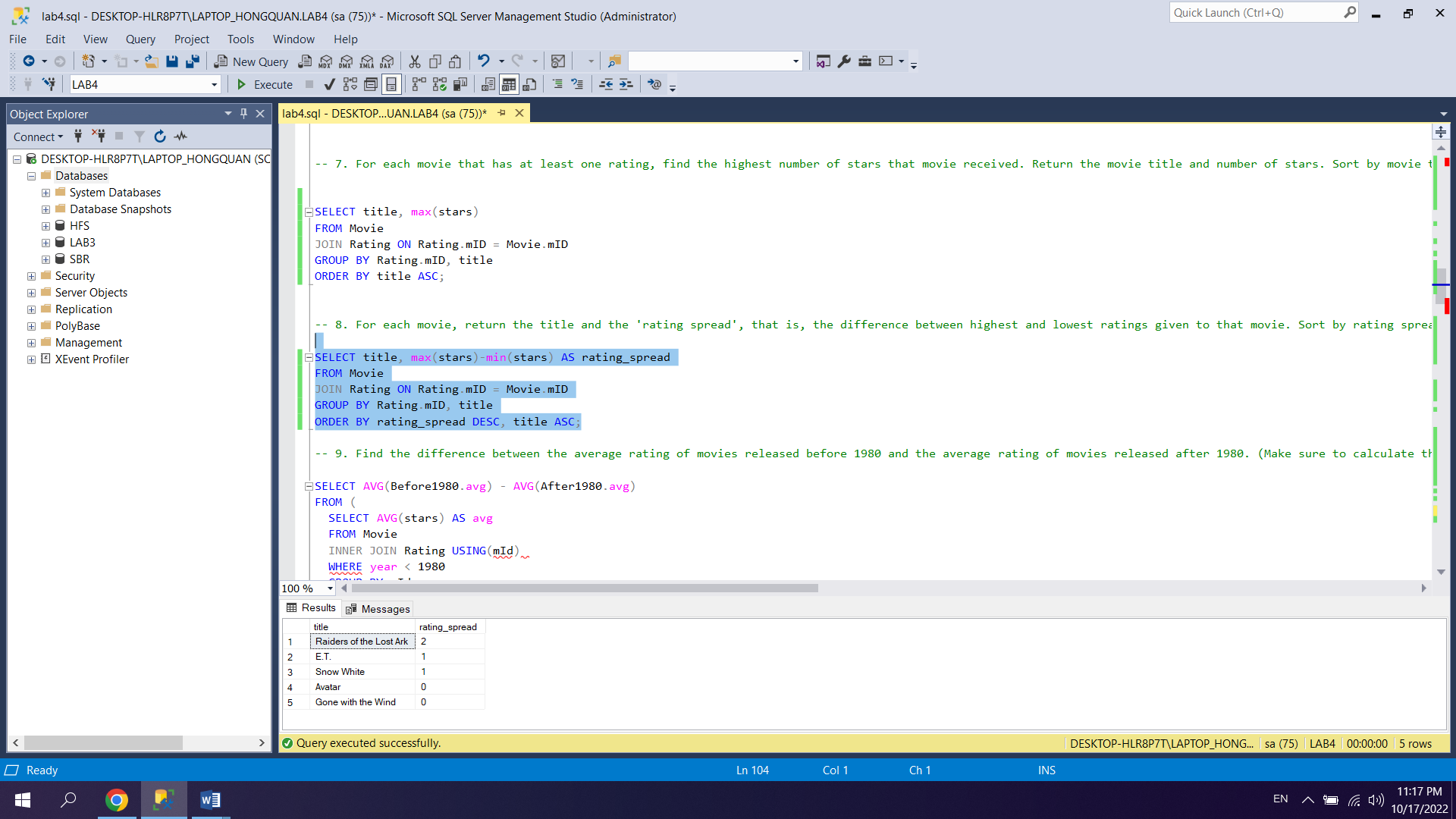
SELECT title, max(stars)-min(stars) AS rating\_spread

FROM Movie

JOIN Rating ON Rating.mID = Movie.mID

GROUP BY Rating.mID, title

ORDER BY rating\_spread DESC, title ASC;



9.

SELECT AVG(Before1980.avg) - AVG(After1980.avg)

FROM (

SELECT AVG(stars) AS avg

FROM Movie

INNER JOIN Rating ON Movie.mID = Rating.mID

WHERE year < 1980

GROUP BY Movie.mID

) AS Before1980, (

SELECT AVG(stars) AS avg

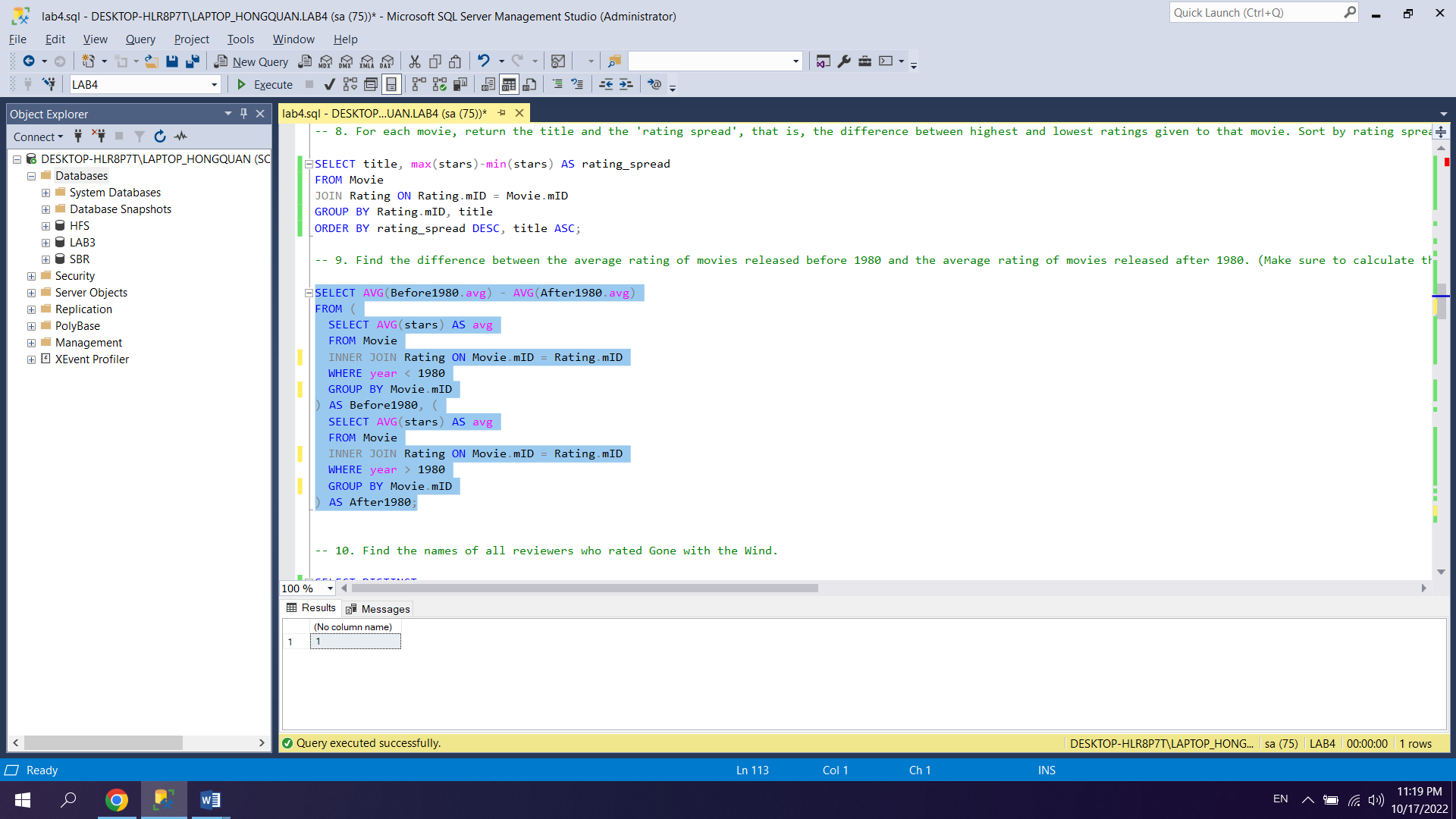
FROM Movie

INNER JOIN Rating ON Movie.mID = Rating.mID

WHERE year > 1980

GROUP BY Movie.mID

) AS After1980;



10.

SELECT DISTINCT name

FROM Rating

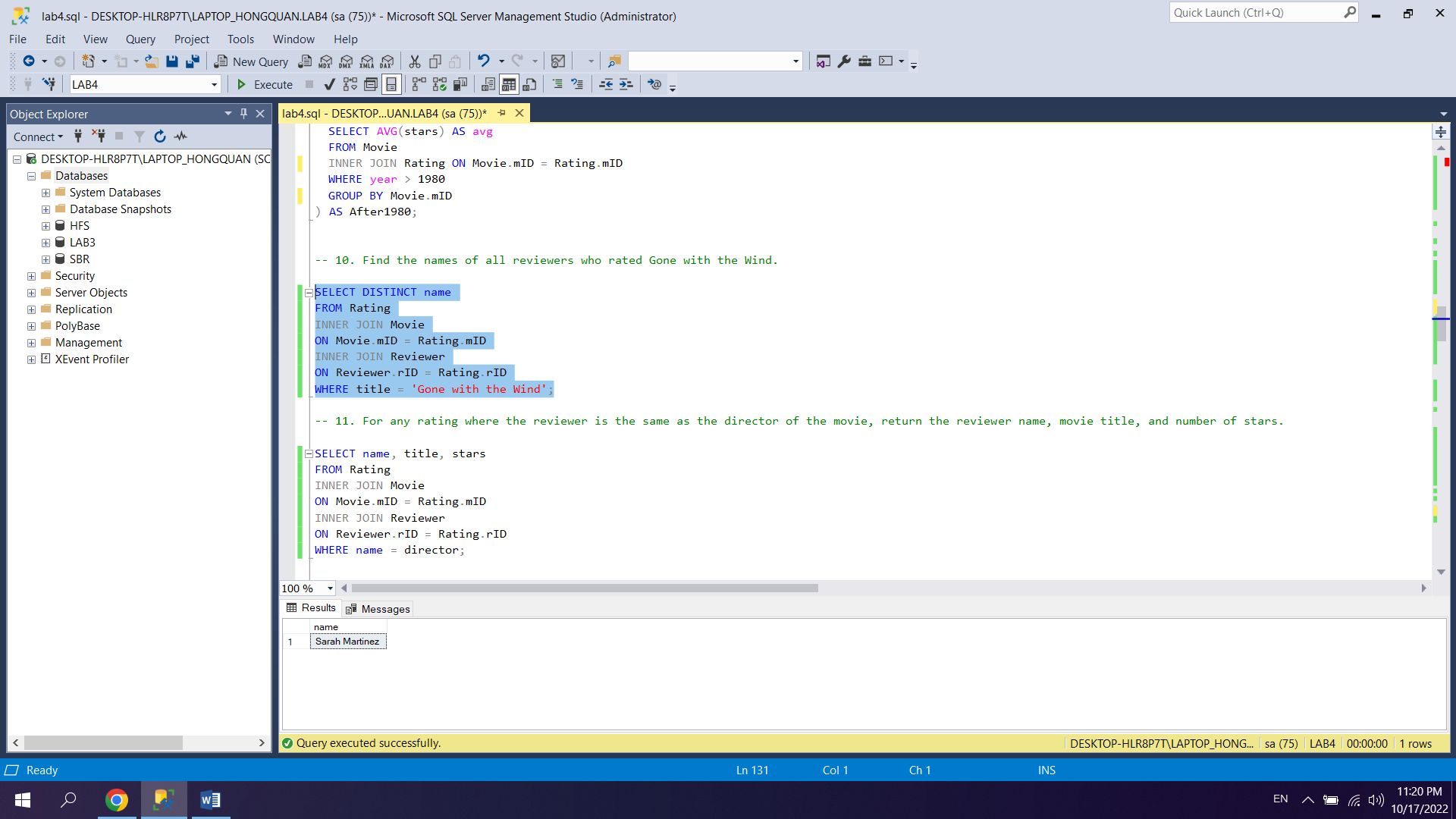
INNER JOIN Movie

ON Movie.mID = Rating.mID

INNER JOIN Reviewer

ON Reviewer.rID = Rating.rID

WHERE title = 'Gone with the Wind';



11.

SELECT name, title, stars

FROM Rating

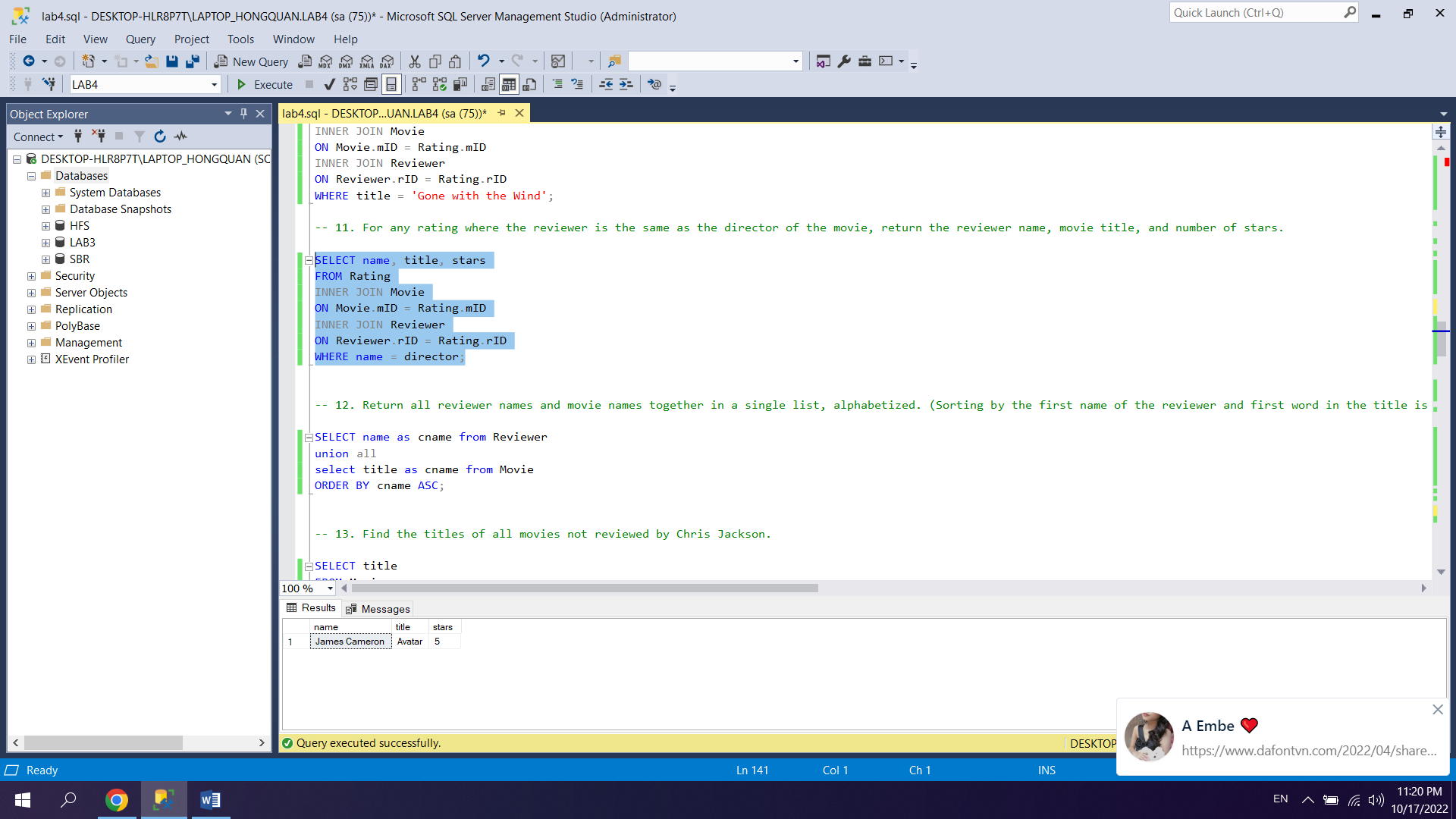
INNER JOIN Movie

ON Movie.mID = Rating.mID

INNER JOIN Reviewer

ON Reviewer.rID = Rating.rID

WHERE name = director;



12.

SELECT name as cname from Reviewer

union all

select title as cname from Movie

ORDER BY cname ASC;



13.

SELECT title

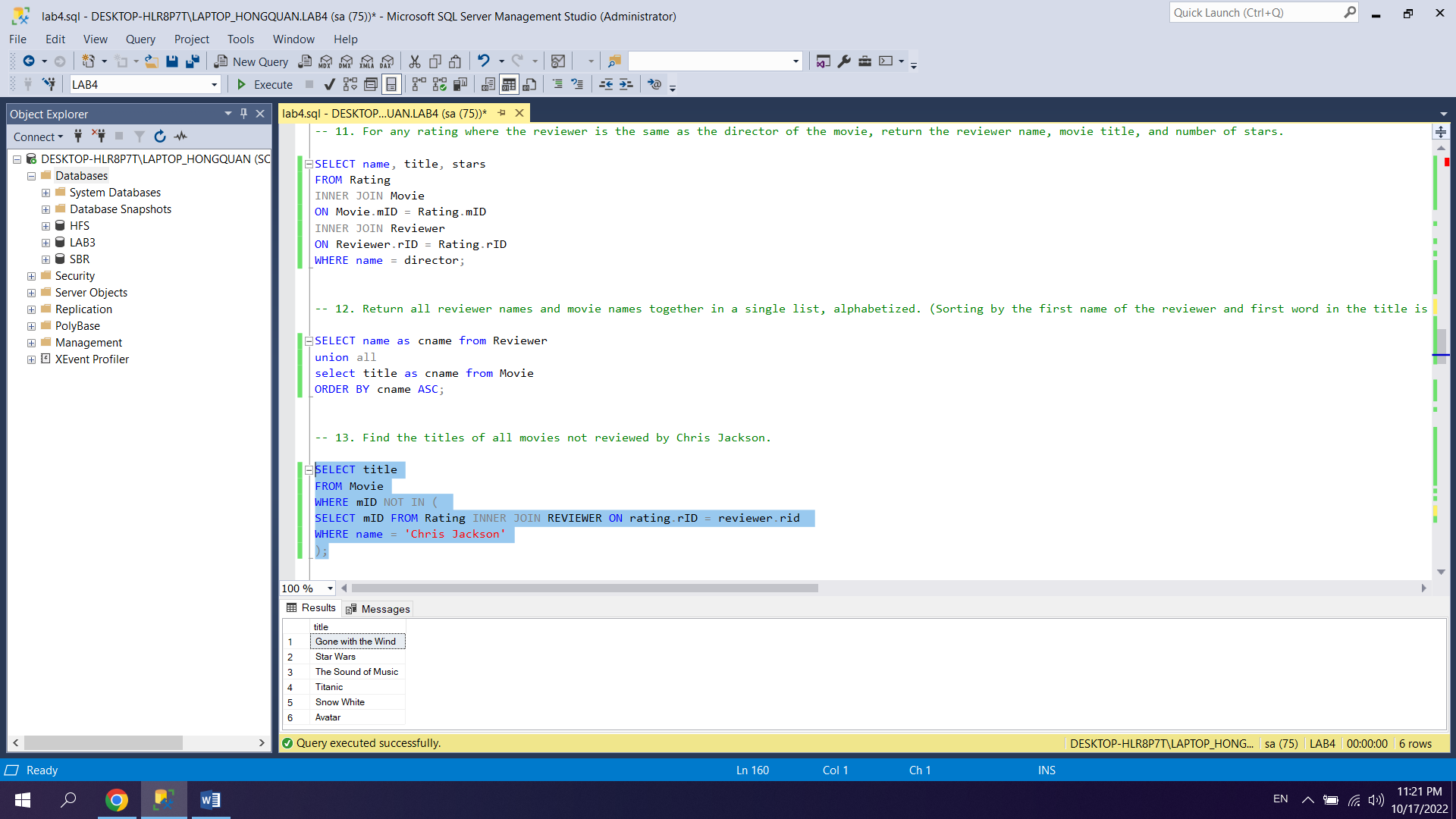
FROM Movie

WHERE mID NOT IN (

SELECT mID FROM Rating INNER JOIN REVIEWER ON rating.rID = reviewer.rid

WHERE name = 'Chris Jackson'

);



14.

SELECT DISTINCT Re1.name, Re2.name

FROM Rating R1, Rating R2, Reviewer Re1, Reviewer Re2

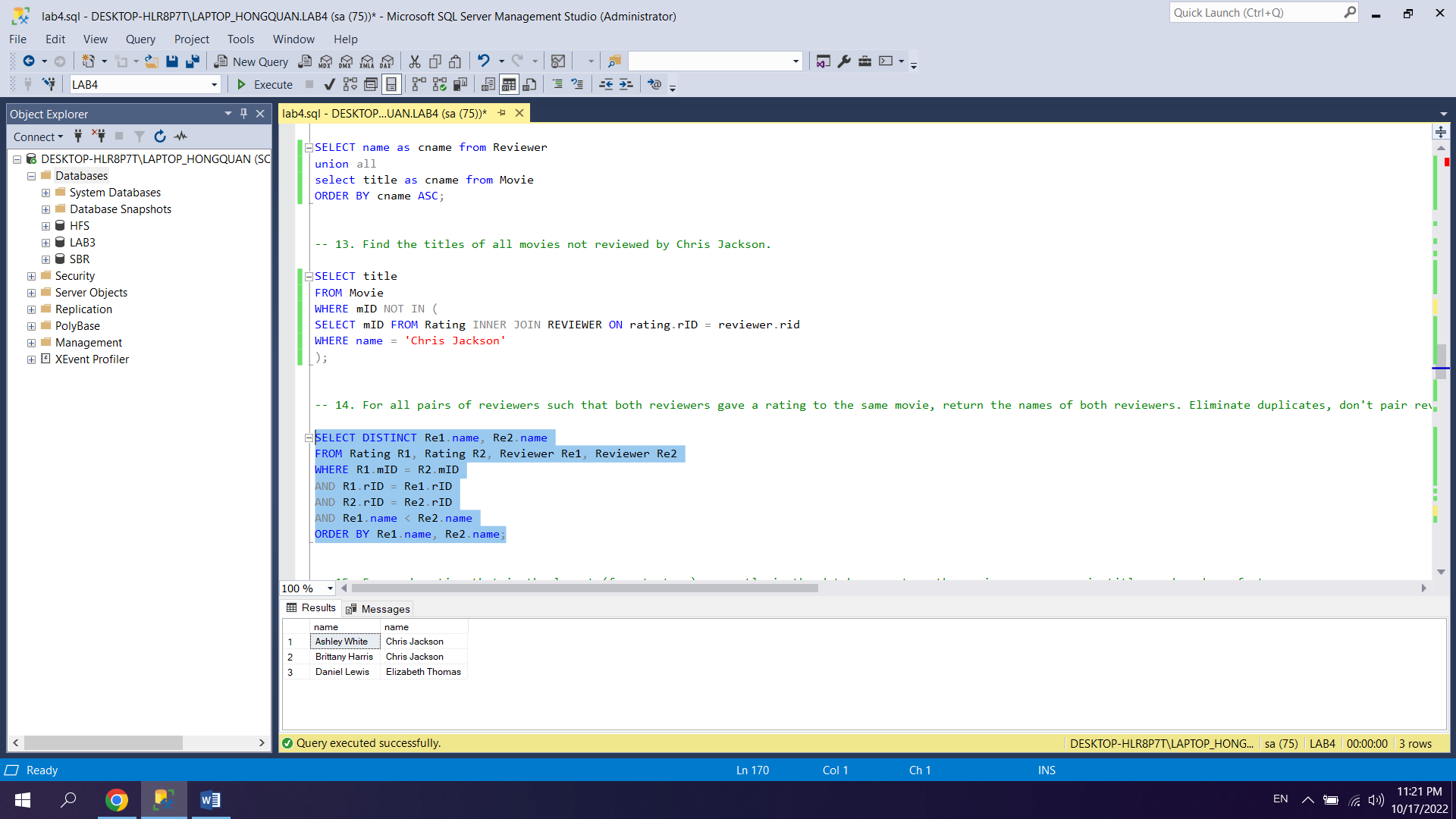
WHERE R1.mID = R2.mID

AND R1.rID = Re1.rID

AND R2.rID = Re2.rID

AND Re1.name < Re2.name

ORDER BY Re1.name, Re2.name;



15.

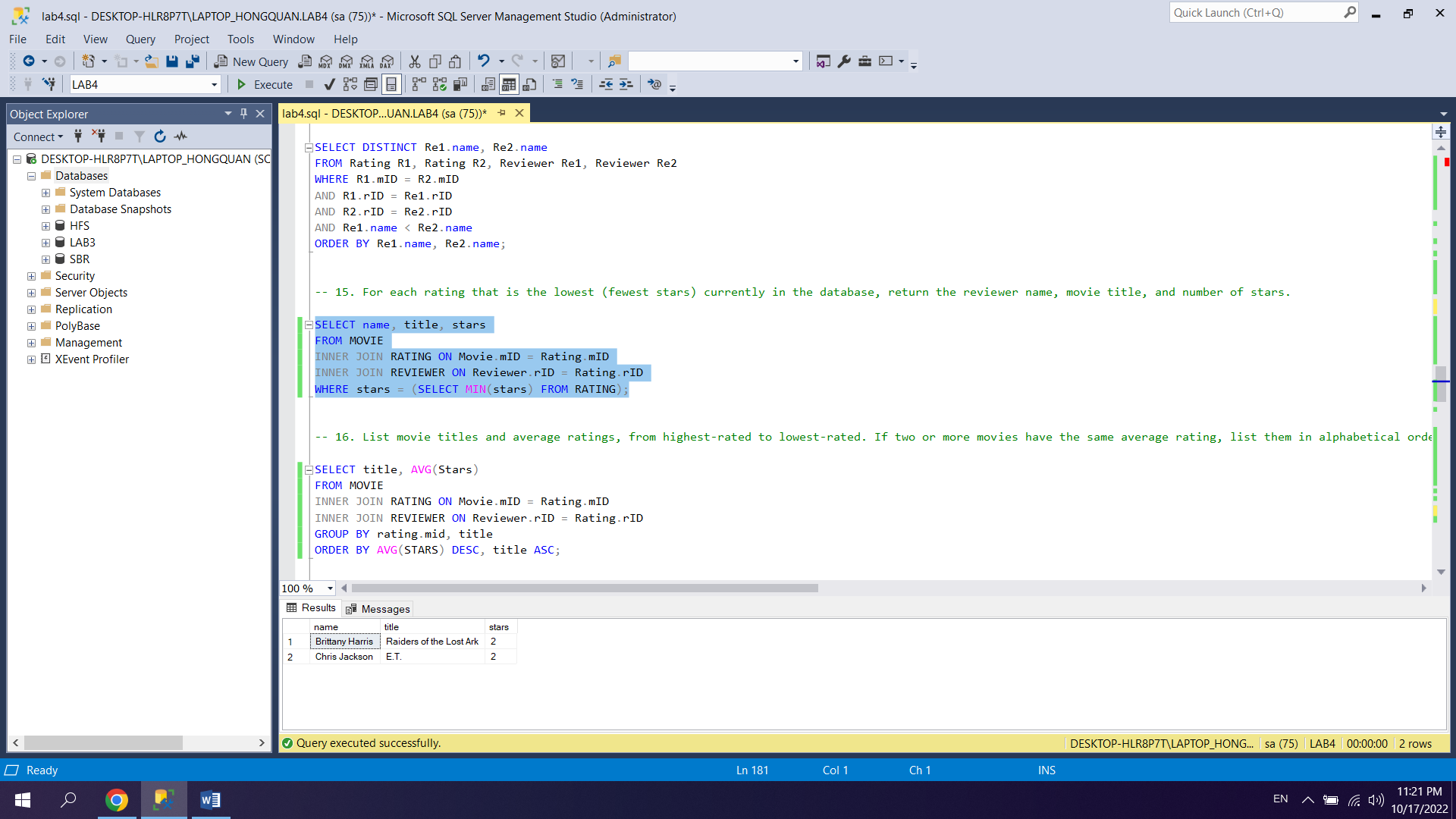
SELECT name, title, stars

FROM MOVIE

INNER JOIN RATING ON Movie.mID = Rating.mID

INNER JOIN REVIEWER ON Reviewer.rID = Rating.rID

WHERE stars = (SELECT MIN(stars) FROM RATING);



16.

SELECT title, AVG(Stars)

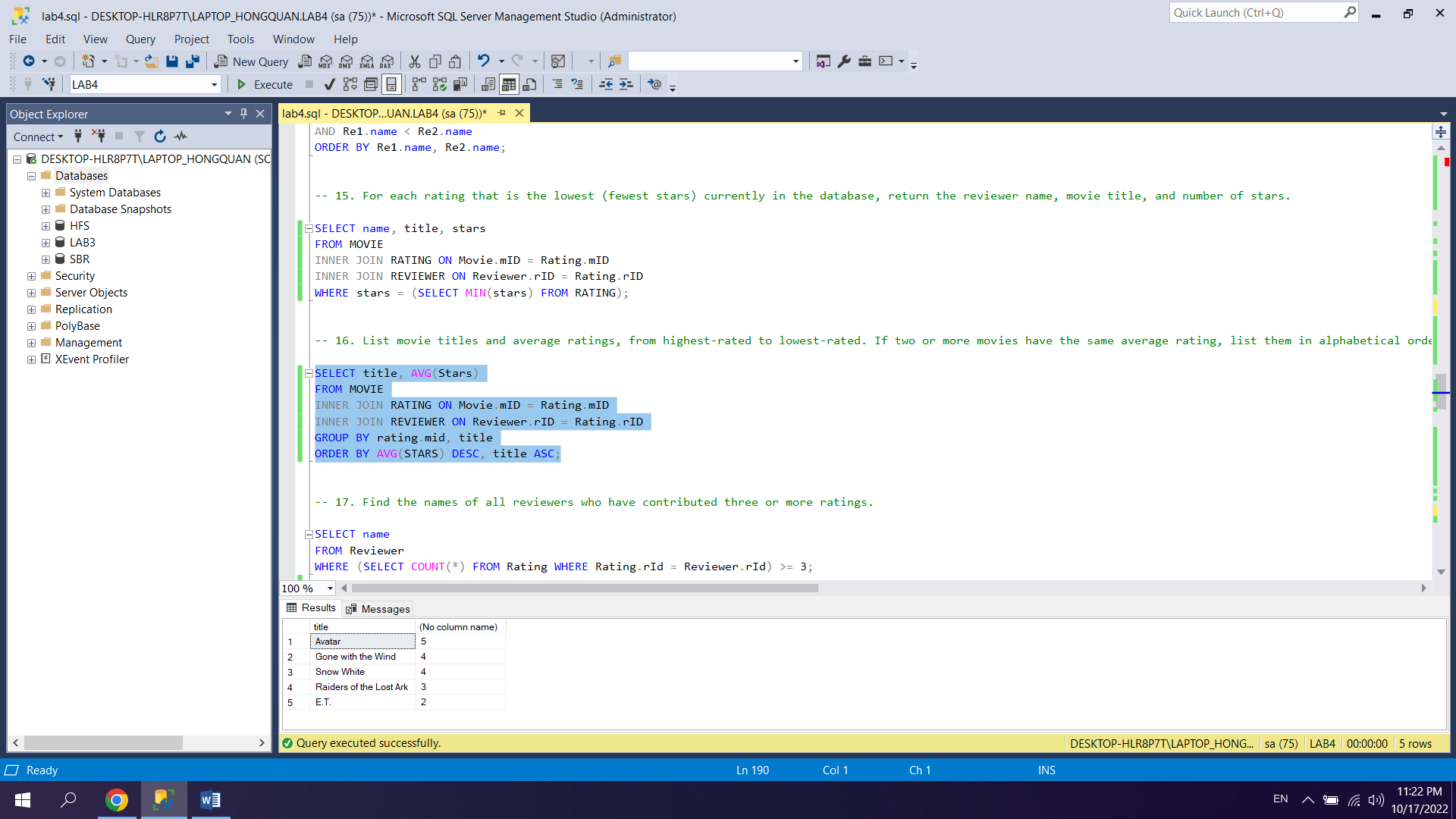
FROM MOVIE

INNER JOIN RATING ON Movie.mID = Rating.mID

INNER JOIN REVIEWER ON Reviewer.rID = Rating.rID

GROUP BY rating.mid, title

ORDER BY AVG(STARS) DESC, title ASC;



17.

SELECT name

FROM Reviewer

WHERE (SELECT COUNT(\*) FROM Rating WHERE Rating.rId = Reviewer.rId) >= 3;



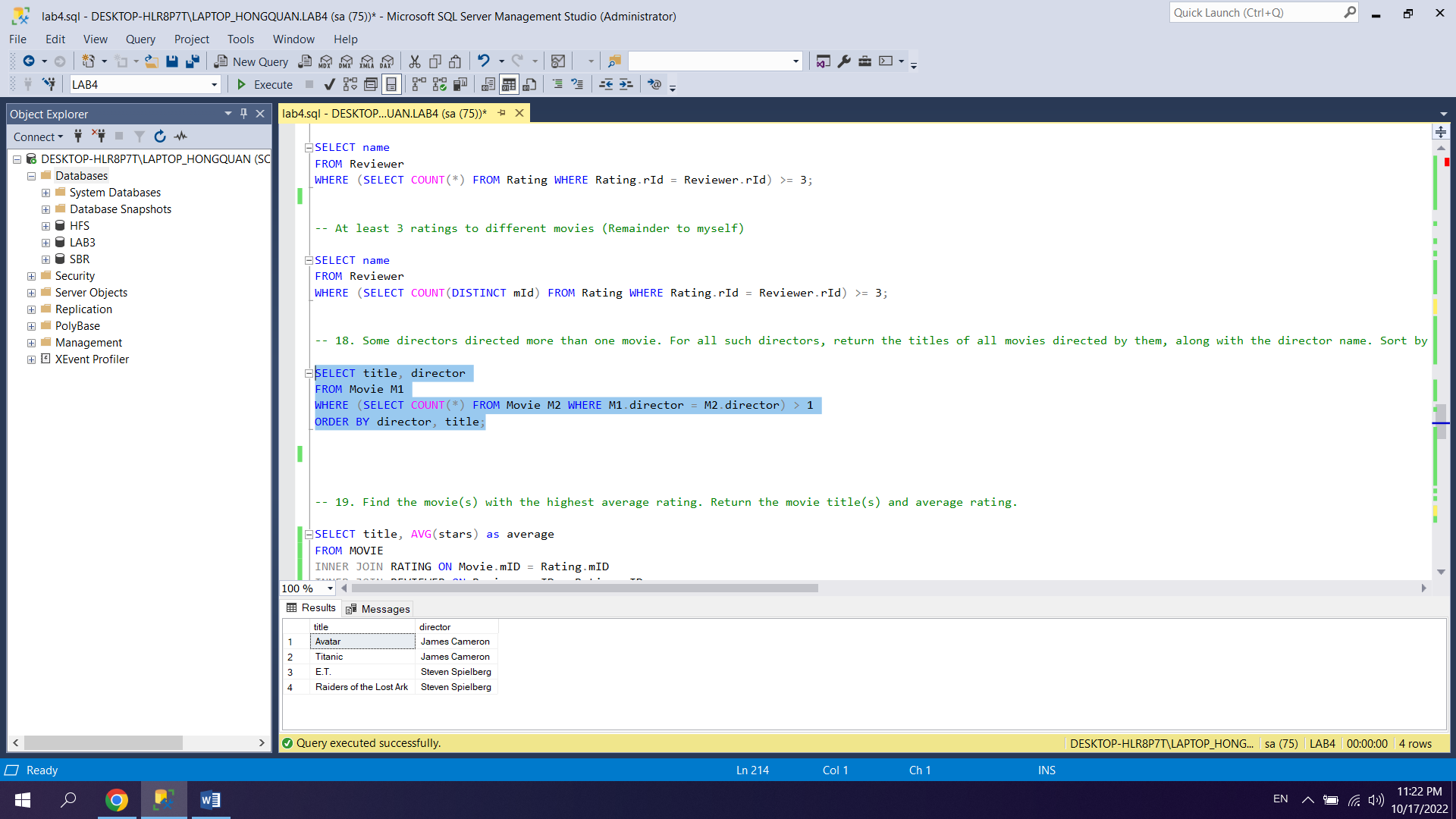
18.

SELECT title, director

FROM Movie M1

WHERE (SELECT COUNT(\*) FROM Movie M2 WHERE M1.director = M2.director) > 1

ORDER BY director, title;



19.

SELECT title, AVG(stars) as average

FROM MOVIE

INNER JOIN RATING ON Movie.mID = Rating.mID

INNER JOIN REVIEWER ON Reviewer.rID = Rating.rID

GROUP BY movie.mID, title

HAVING AVG(stars) = (

SELECT MAX(avg\_stars)

FROM (

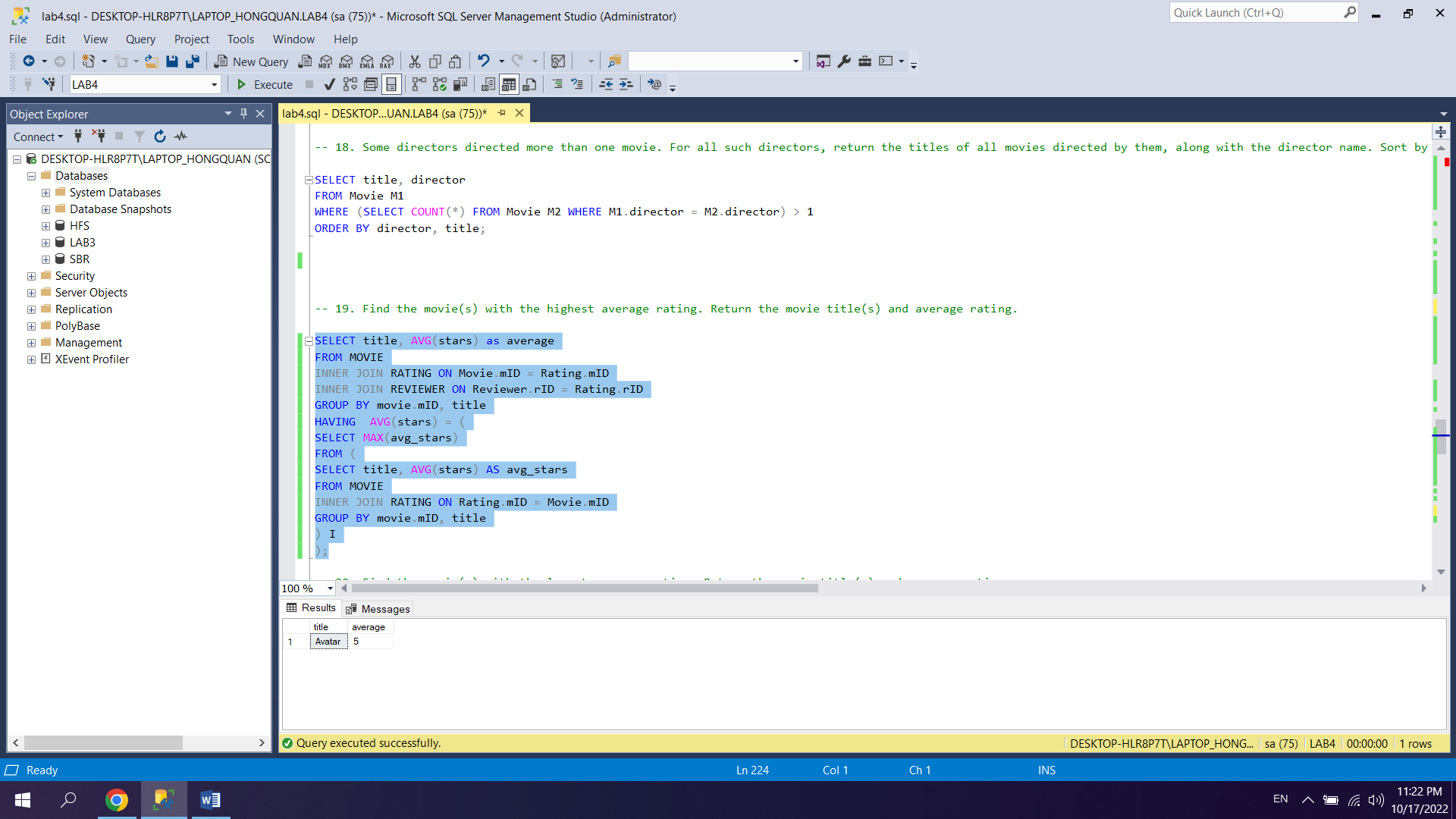
SELECT title, AVG(stars) AS avg\_stars

FROM MOVIE

INNER JOIN RATING ON Rating.mID = Movie.mID

GROUP BY movie.mID, title

) I

);

20.

SELECT title, AVG(stars) as average

FROM MOVIE

INNER JOIN RATING ON Movie.mID = Rating.mID

INNER JOIN REVIEWER ON Reviewer.rID = Rating.rID

GROUP BY movie.mID, title

HAVING AVG(stars) = (

SELECT MIN(avg\_stars)

FROM (

SELECT title, AVG(stars) AS avg\_stars

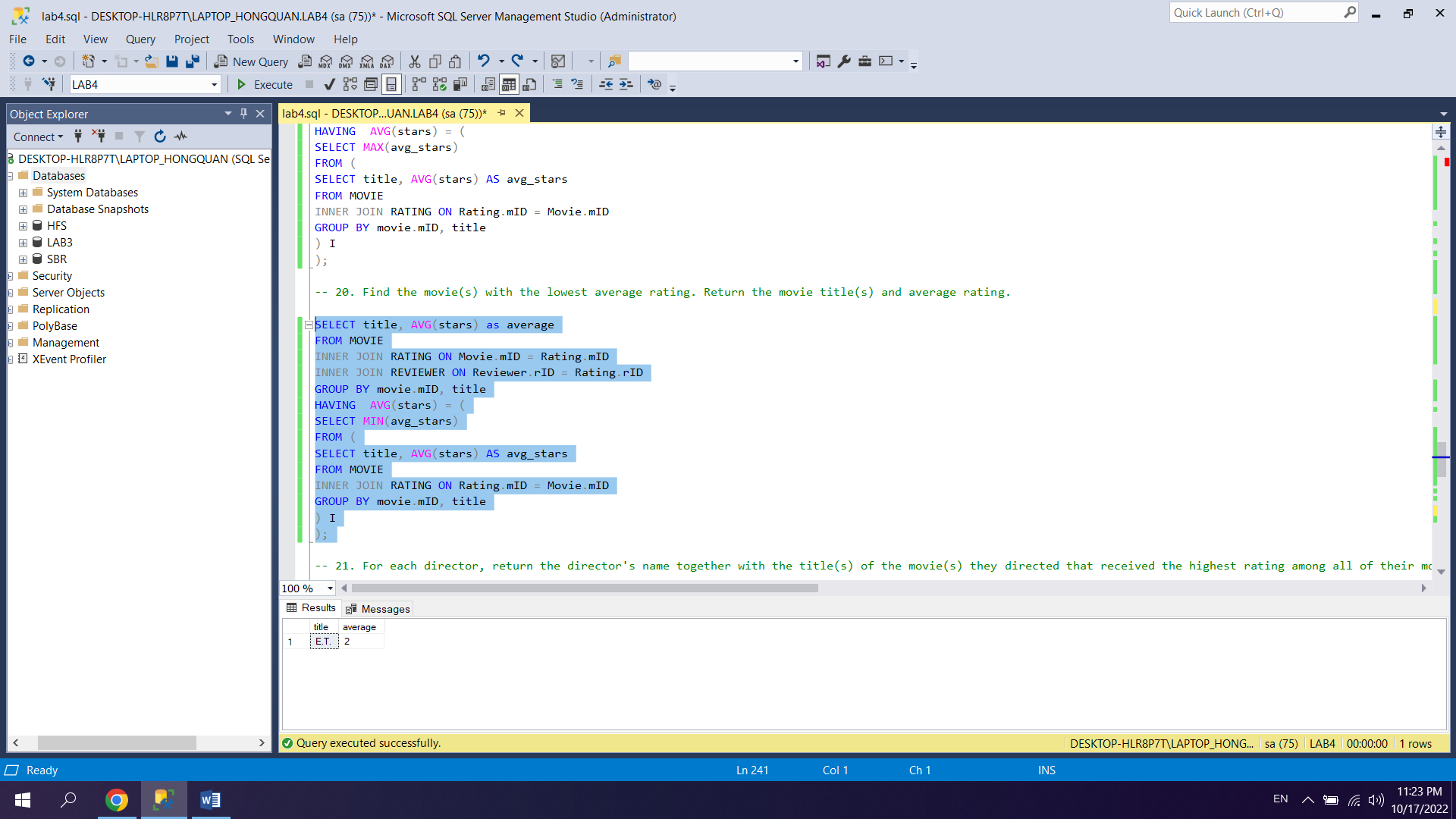
FROM MOVIE

INNER JOIN RATING ON Rating.mID = Movie.mID

GROUP BY movie.mID, title

) I

);



21.

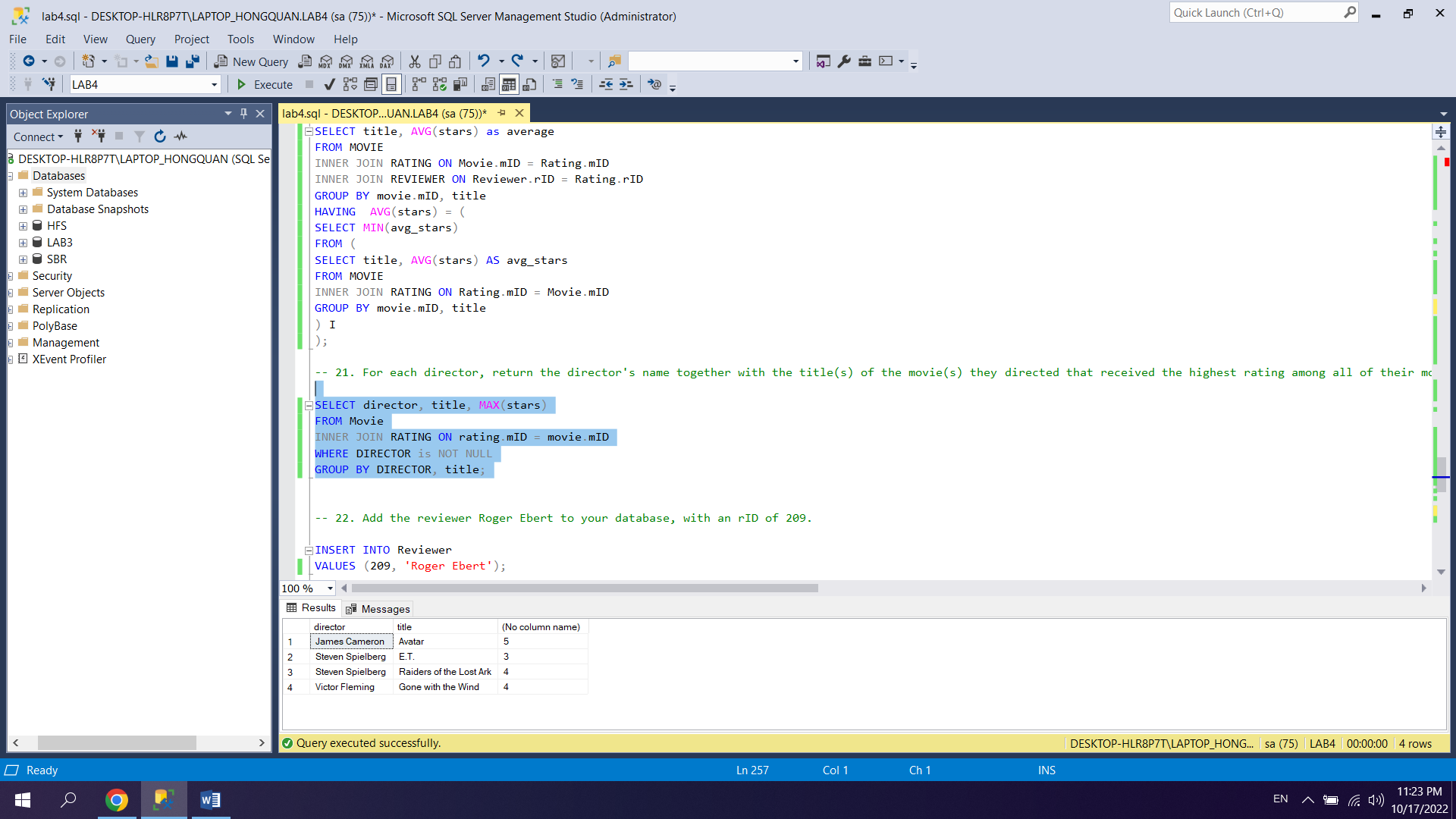
SELECT director, title, MAX(stars)

FROM Movie

INNER JOIN RATING ON rating.mID = movie.mID

WHERE DIRECTOR is NOT NULL

GROUP BY DIRECTOR, title;



22.

INSERT INTO Reviewer

VALUES (209, 'Roger Ebert');

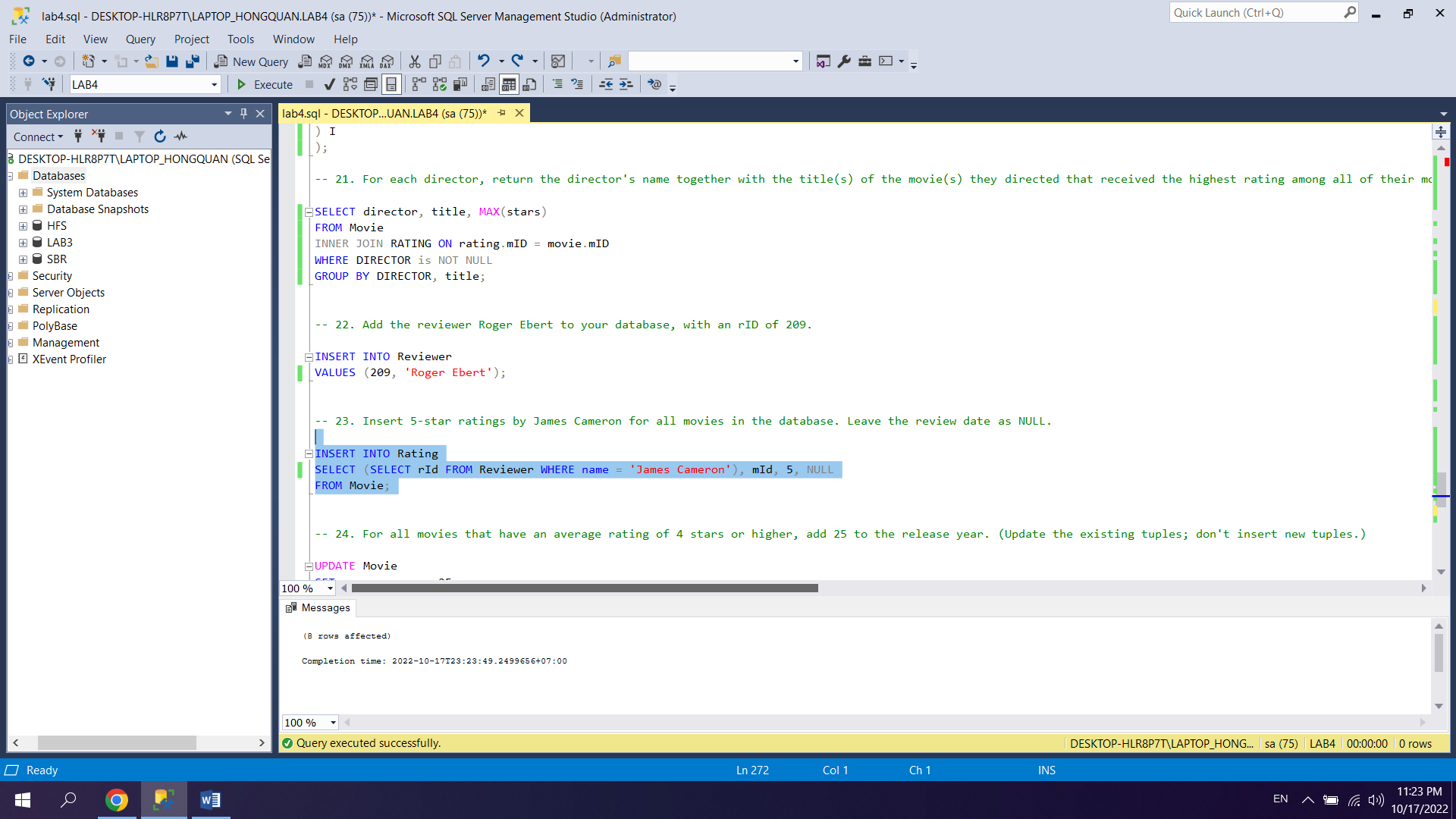


23.

INSERT INTO Rating

SELECT (SELECT rId FROM Reviewer WHERE name = 'James Cameron'), mId, 5, NULL

FROM Movie;



24.

UPDATE Movie

SET year = year + 25

WHERE Movie.mID IN (

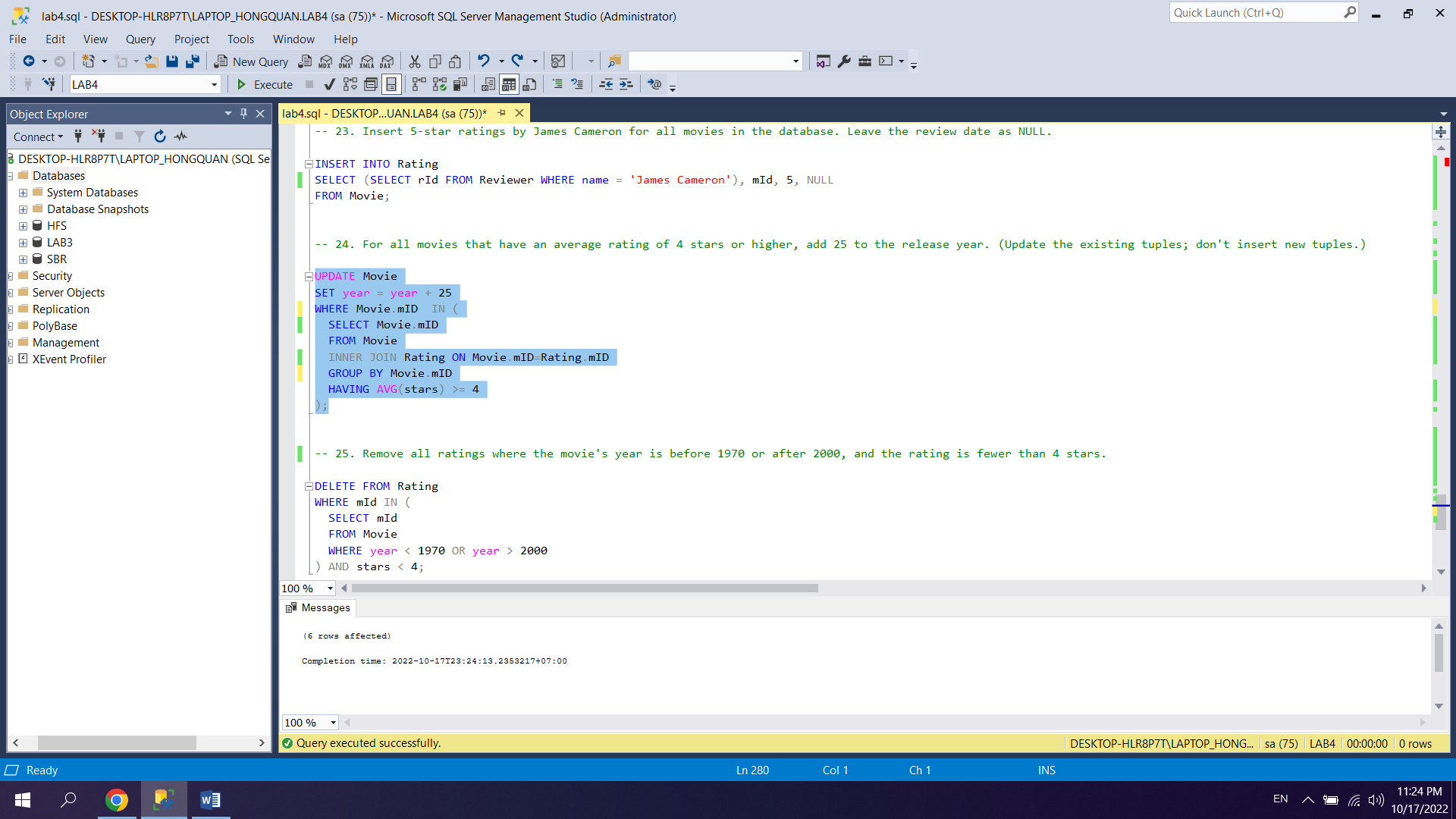
SELECT Movie.mID

FROM Movie

INNER JOIN Rating ON Movie.mID=Rating.mID

GROUP BY Movie.mID

HAVING AVG(stars) >= 4

);

25.

DELETE FROM Rating

WHERE mId IN (

SELECT mId

FROM Movie

WHERE year < 1970 OR year > 2000

) AND stars < 4;

