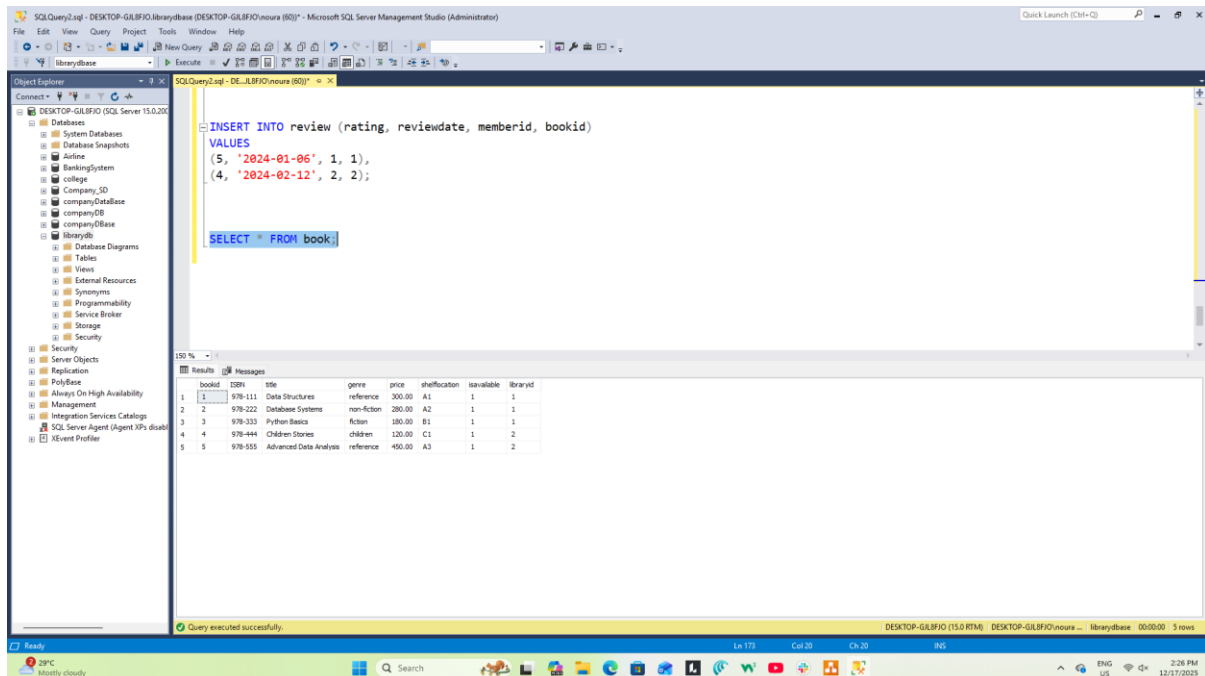


Library Database – DQL & DML Tasks

DQL

1. Display all book records.



The screenshot shows the SQL Server Enterprise interface. The query window contains the following SQL code:

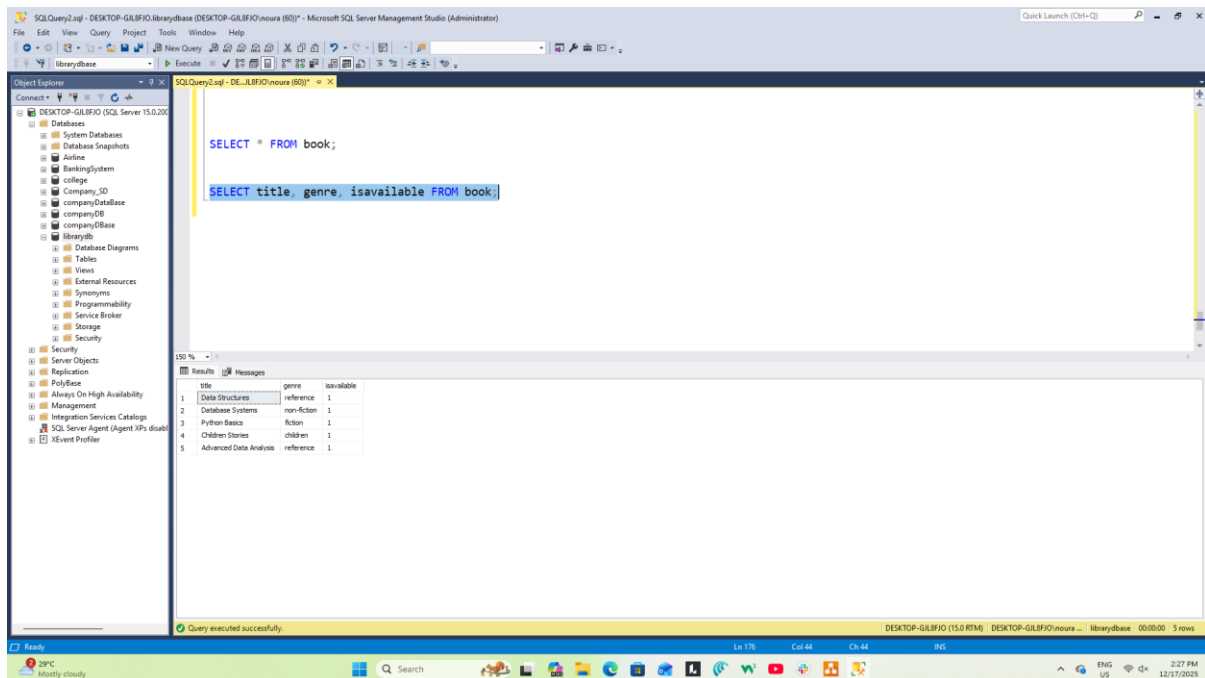
```
INSERT INTO review (rating, reviewdate, memberid, bookid)
VALUES
(5, '2024-01-06', 1, 1),
(4, '2024-02-12', 2, 2);

SELECT * FROM book;
```

The query was executed successfully, and the results are displayed in a table with the following data:

bookid	ISBN	title	genre	price	shelflocation	isavailable	libraryid
1	978-111	Data Structures	reference	300.00	A1	1	1
2	978-222	Database Systems	non-fiction	280.00	A2	1	1
3	978-333	Python Basics	fiction	180.00	B1	1	1
4	978-444	Children Stories	children	120.00	C1	1	2
5	978-555	Advanced Data Analysis	reference	400.00	A3	1	2

2. Display each book's title, genre, and availability.



The screenshot shows the SQL Server Enterprise interface. The query window contains the following SQL code:

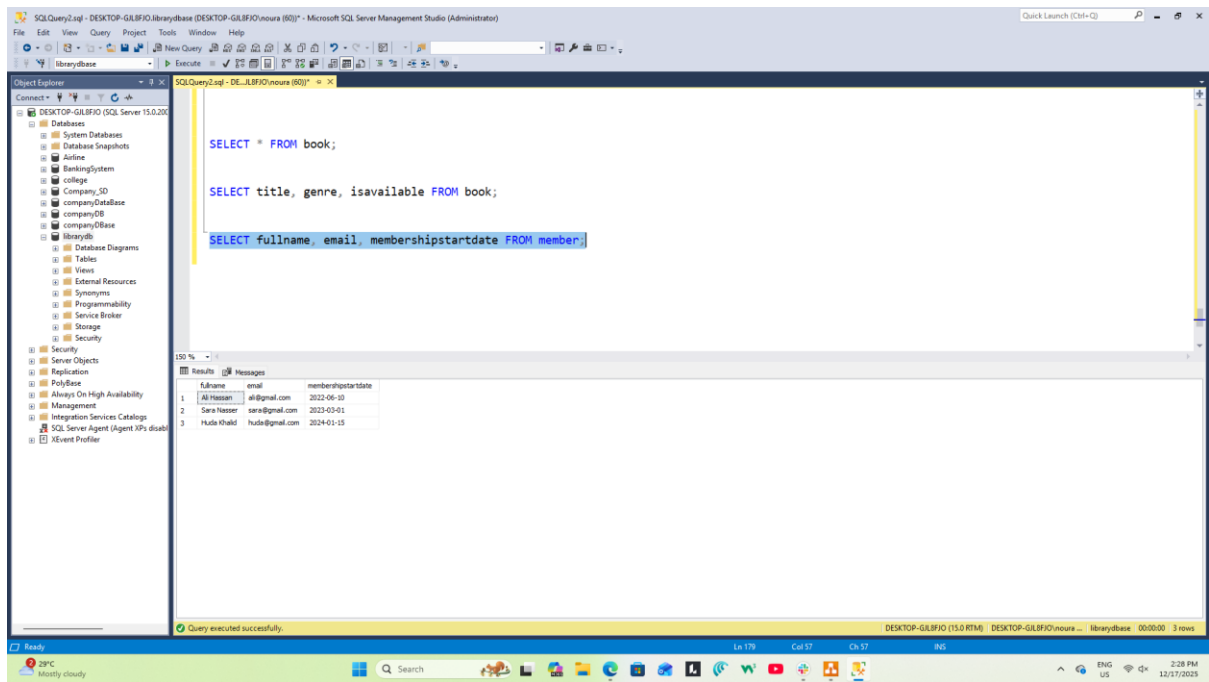
```
SELECT * FROM book;

SELECT title, genre, isavailable FROM book;
```

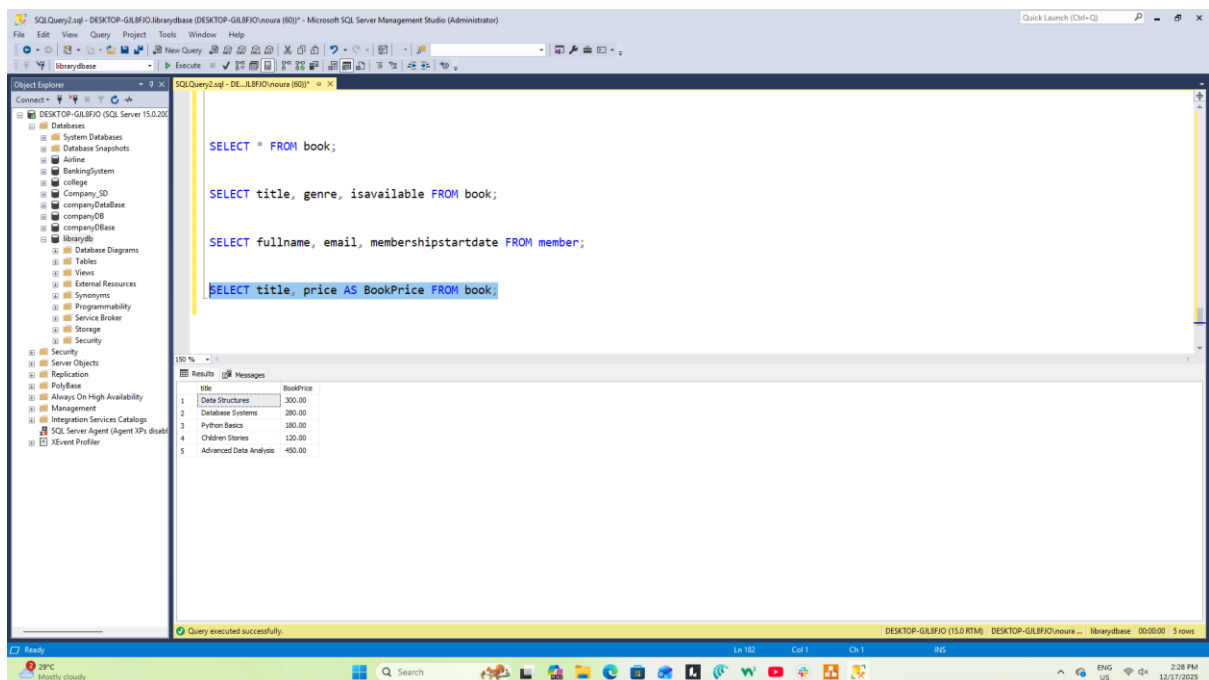
The query was executed successfully, and the results are displayed in a table with the following data:

title	genre	isavailable
Data Structures	reference	1
Database Systems	non-fiction	1
Python Basics	fiction	1
Children Stories	children	1
Advanced Data Analysis	reference	1

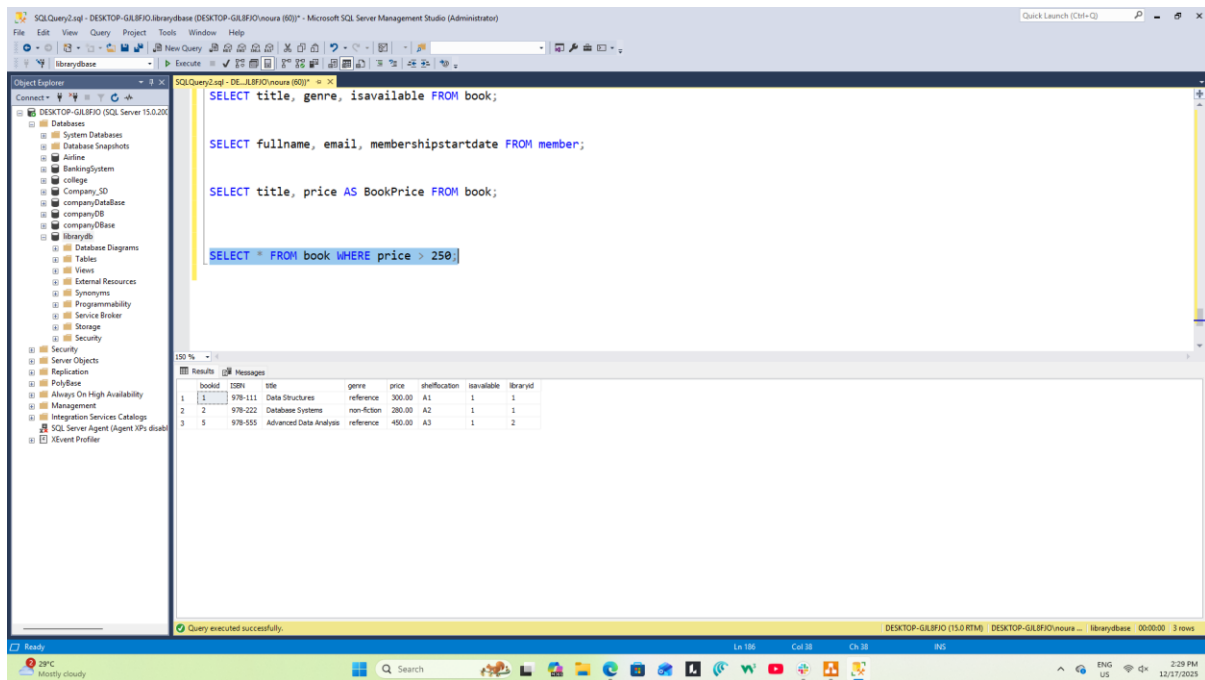
3. Display all member names, email, and membership start date.



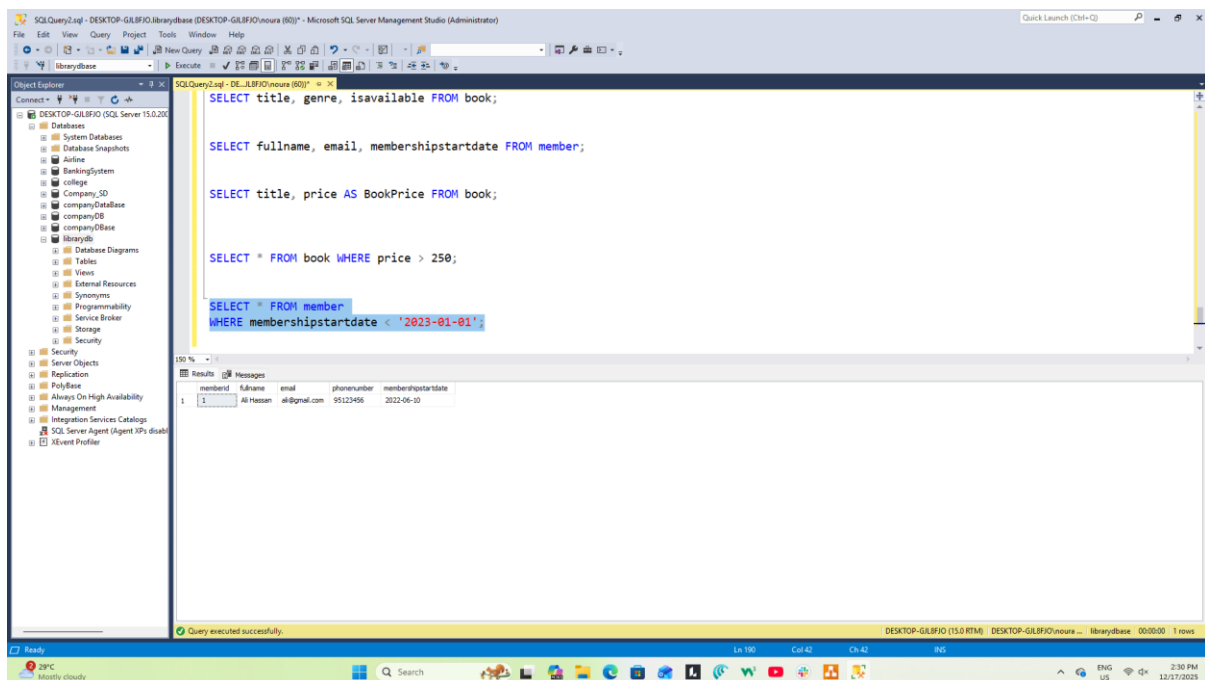
4. Display each book's title and price as BookPrice.



5. List books priced above 250 LE.



6. List members who joined before 2023.



7. Display books published after 2018.

`SELECT *FROM book WHERE publishedyear > 2018;`

The Book table does not contain an attribute that represents the publication year of a book. Therefore, the required query cannot be performed.

Modify the table structure
Add a new column to store the publication year:

```
ALTER TABLE book
```

```
ADD publishedyear INT;
```

Insert or update publication year values

Assign publication years to existing records:

```
UPDATE book
```

```
SET publishedyear = 2020
```

```
WHERE bookid = 1;
```

```
UPDATE book
```

```
SET publishedyear = 2017
```

```
WHERE bookid = 2;
```

(Repeat as needed for other books.)

Execute the required query

Now the query can be executed correctly:

```
SELECT *
```

```
FROM book
```

```
WHERE publishedyear > 2018;
```

Conclusion

By adding the PublishedYear attribute to the Book table and populating it with valid data, the database can support queries that retrieve books published after a specific year.

8. Display books ordered by price descending.

The screenshot shows the SQL Server Enterprise Manager interface. The query window contains the following SQL queries:

```

SELECT * FROM member
WHERE membershipstartdate < '2023-01-01';

SELECT * FROM book WHERE publishedyear > 2018;

SELECT * FROM book ORDER BY price DESC;

```

The results pane shows the output of the third query, displaying a table with 5 rows and 8 columns:

bookid	ISBN	title	genre	price	shelflocation	isavailable	libraryid
5	978-555	Advanced Data Analysis	reference	450.00	A3	1	2
1	978-111	Data Structures	reference	300.00	A1	1	1
2	978-222	Database Systems	non-fiction	200.00	A2	1	1
3	978-333	Python Basics	fiction	180.00	B1	1	1
4	978-444	Children Stories	children	120.00	C1	1	2

9. Display the maximum, minimum, and average book price.

The screenshot shows the SQL Server Enterprise Manager interface. The query window contains the following SQL query:

```

SELECT * FROM book ORDER BY price DESC;

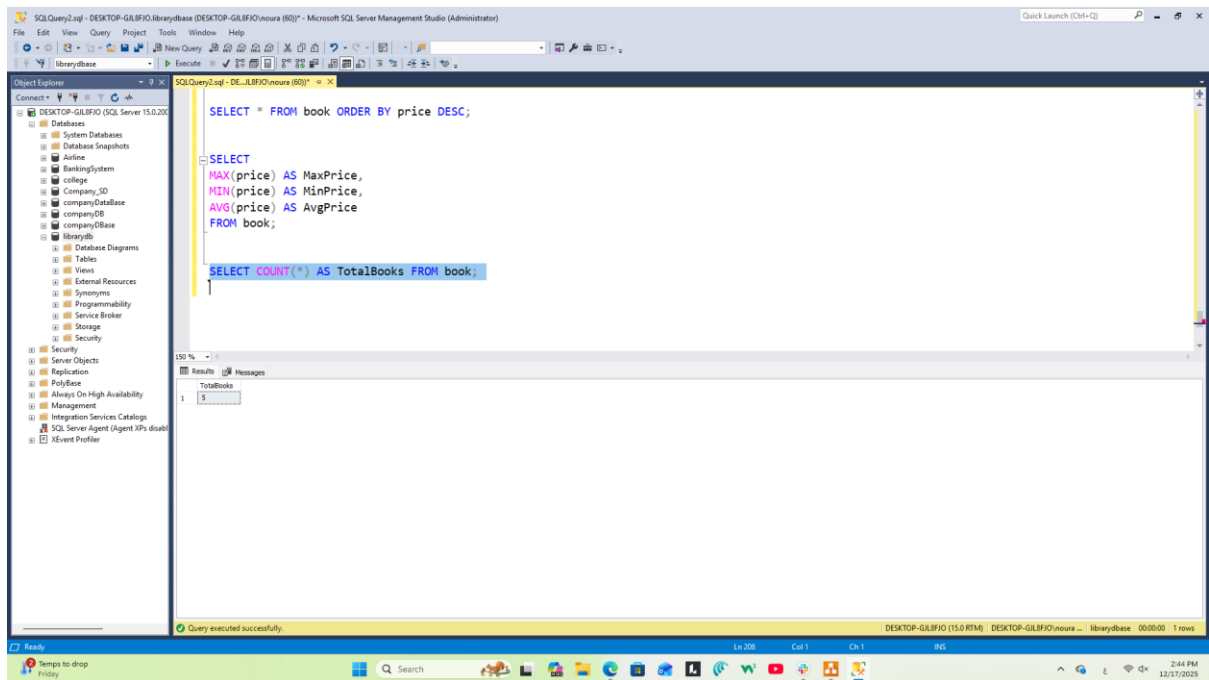
SELECT
    MAX(price) AS MaxPrice,
    MIN(price) AS MinPrice,
    AVG(price) AS AvgPrice
FROM book;

```

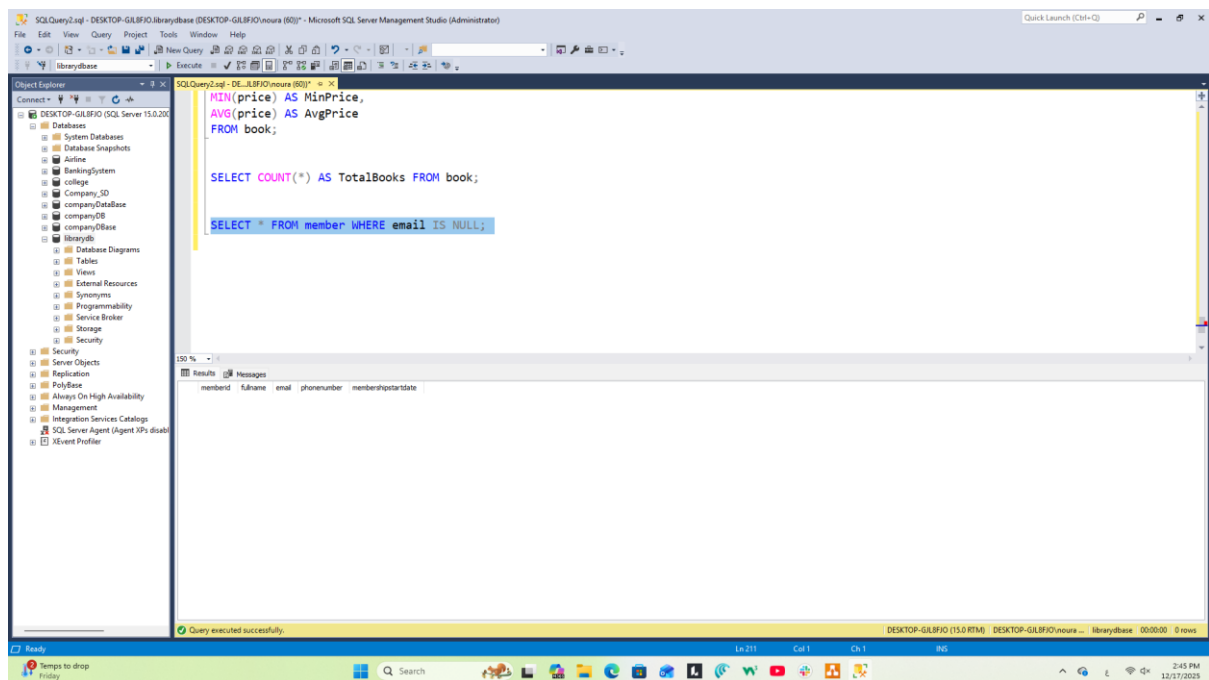
The results pane shows the output of the second query, displaying a table with 1 row and 3 columns:

MaxPrice	MinPrice	AvgPrice
450.00	120.00	266.000000

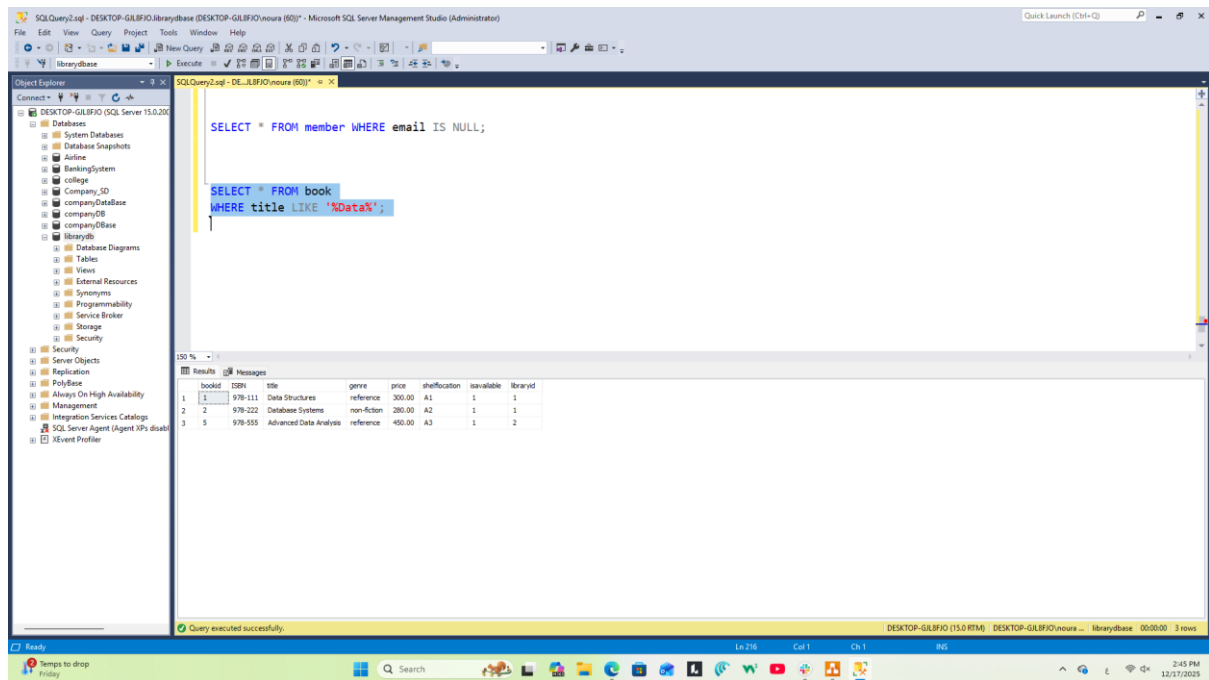
10. Display total number of books



11. Display members with NULL email.

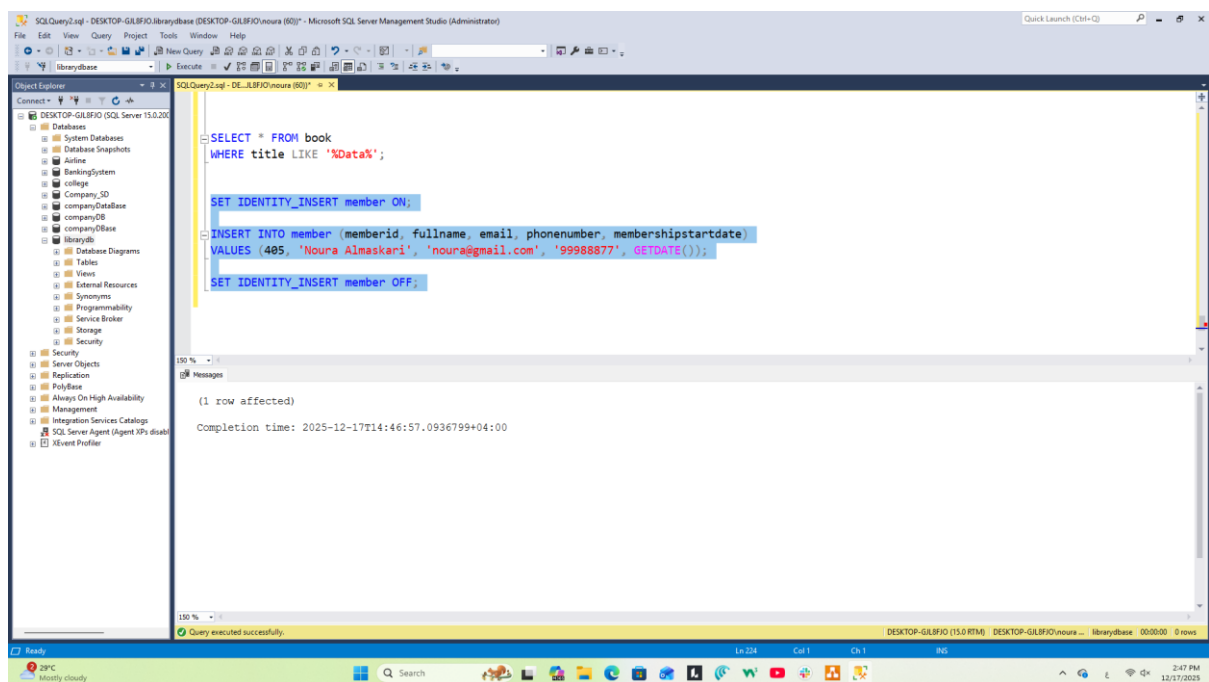


12. Display books whose title contains 'Data'.

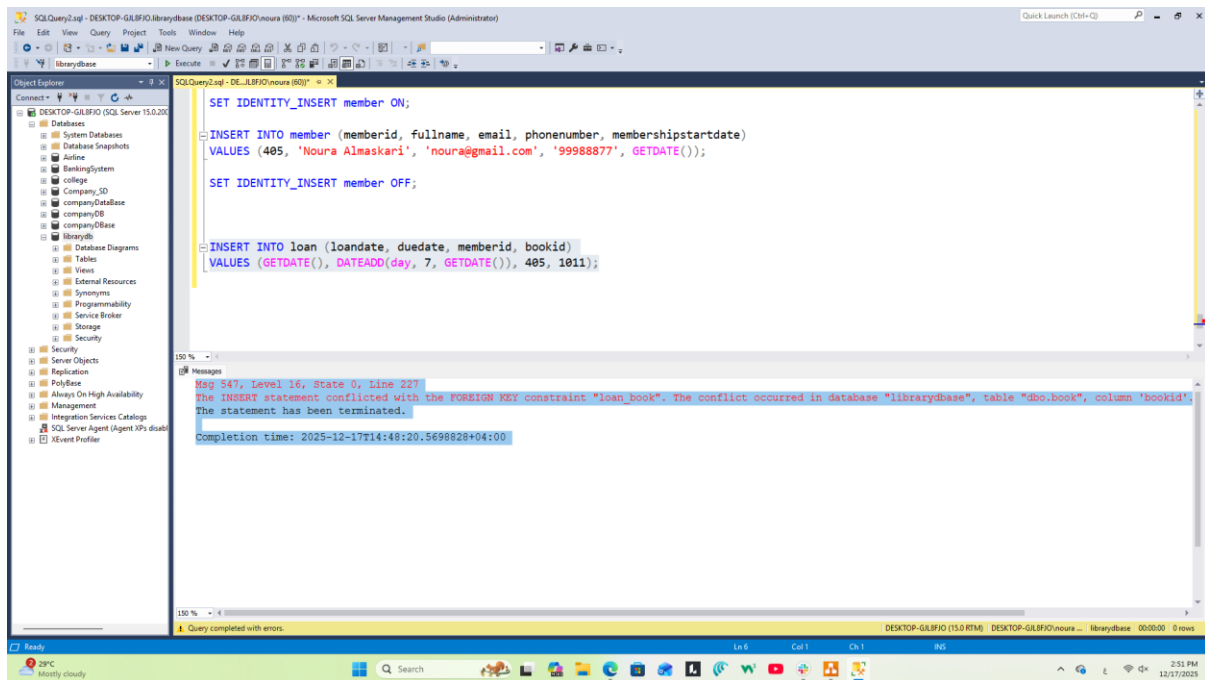


DML

13. Insert yourself as a member (Member ID = 405).

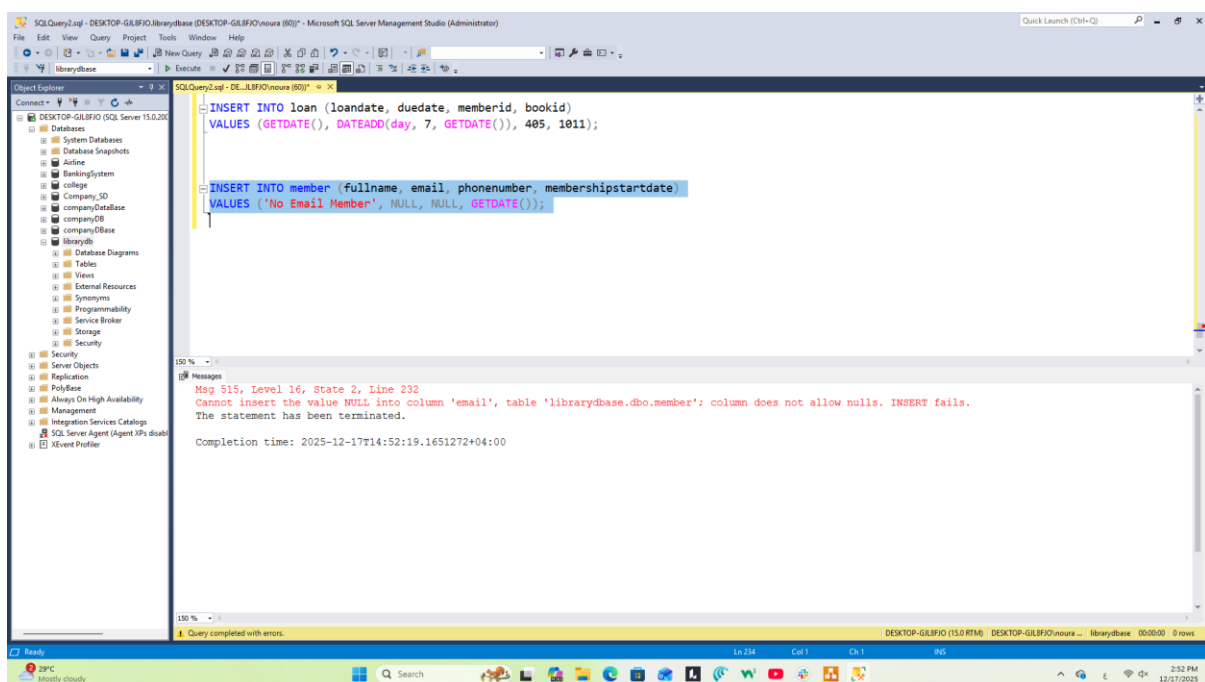


14. Register yourself to borrow book ID 1011.



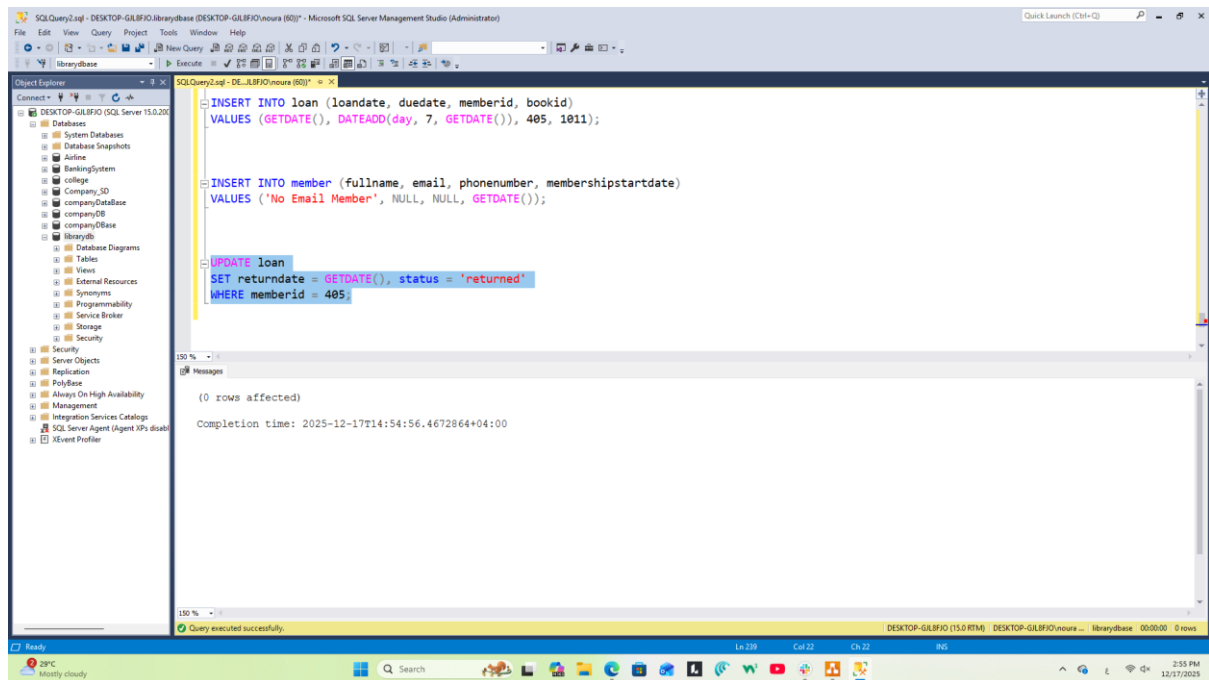
The INSERT statement failed because the referenced BookID does not exist in the Book table.

15. Insert another member with NULL email and phone.

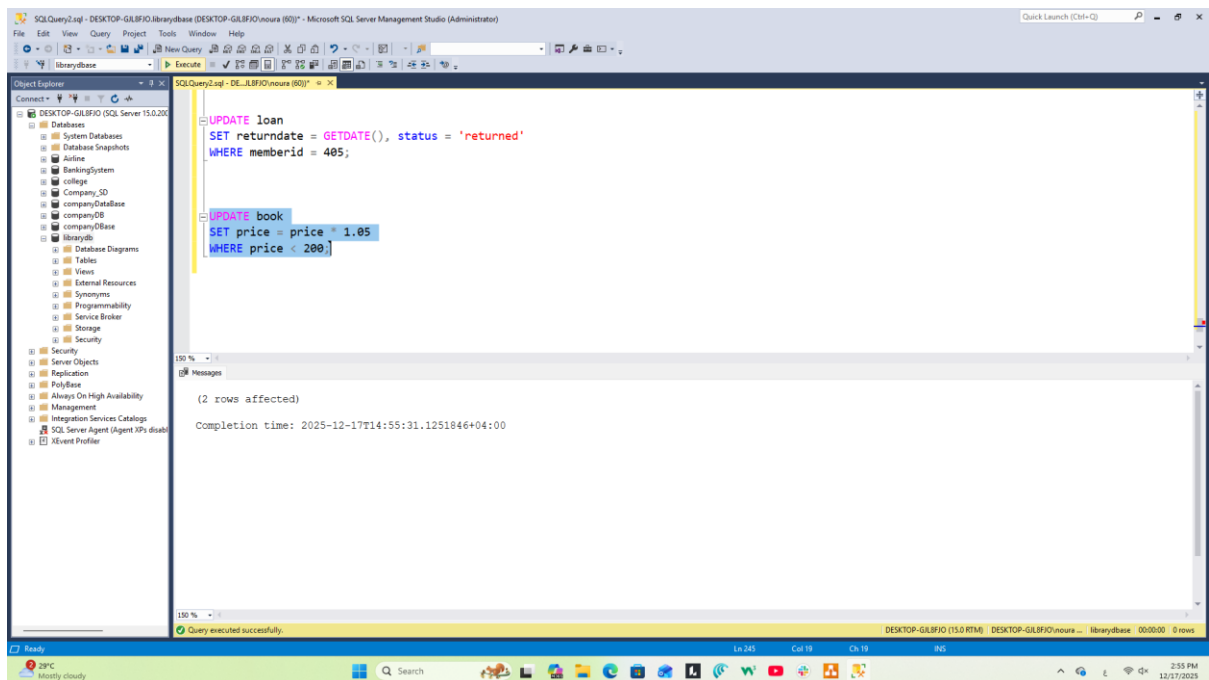


This operation is not possible because the Email attribute in the Member table does not allow NULL values according to the table constraints.

16. Update the return date of your loan to today.

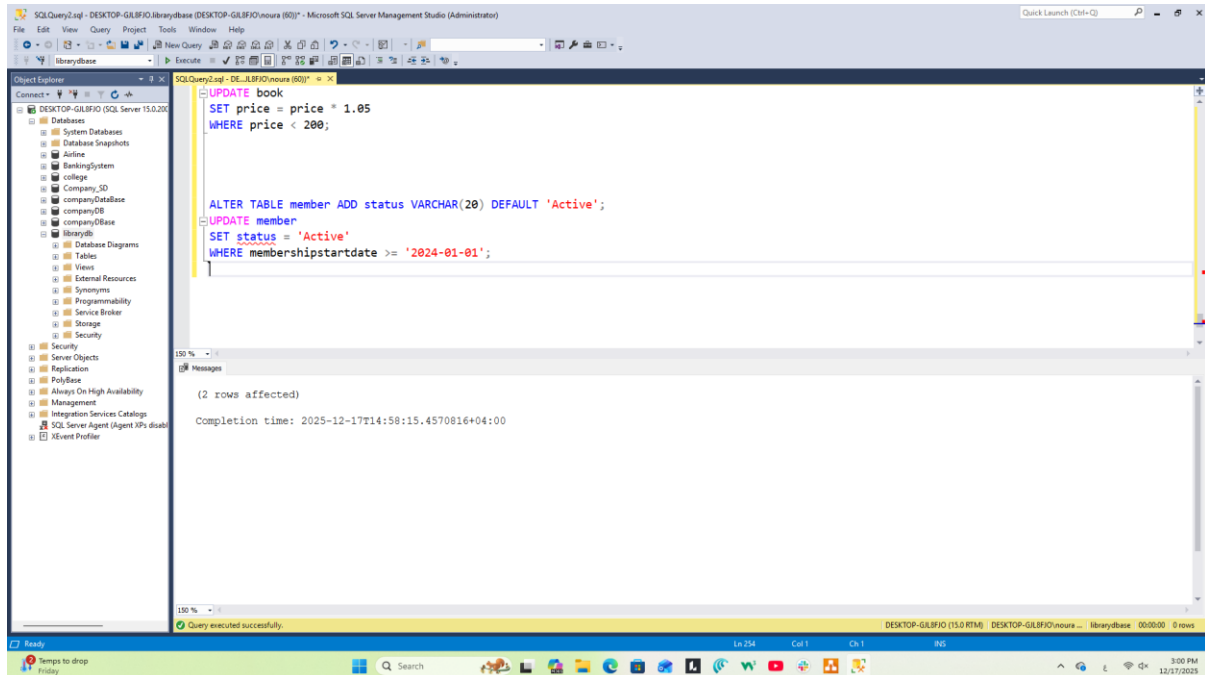


17. Increase book prices by 5% for books priced under 200.



18. Update member status to 'Active' for recently joined members.

The original table design does not include a status column in the Member table.
Therefore, it is not possible to directly update a member's status unless the column is added first.



19. Delete members who never borrowed a book.

