

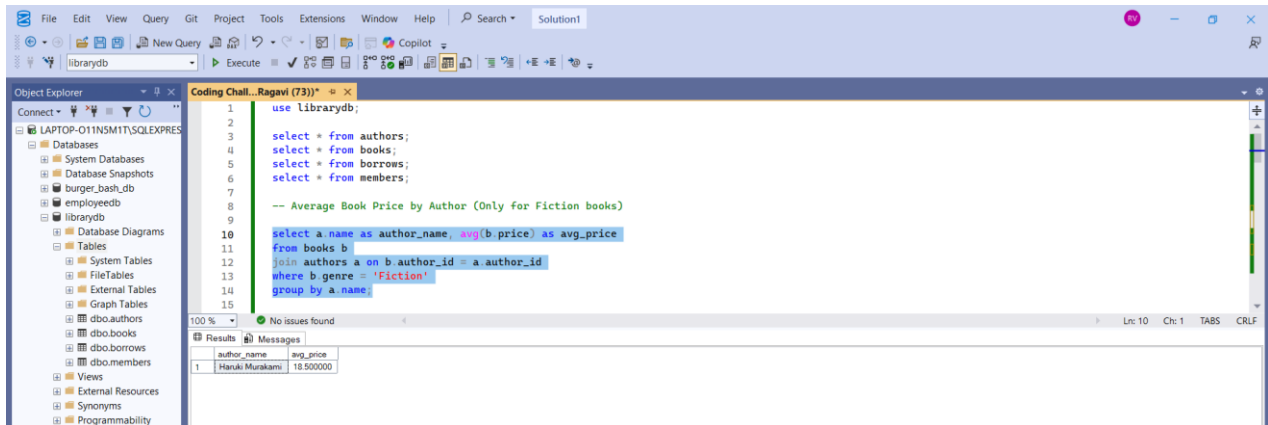
SQL Coding Challenge Documentation

Database: librarydb

1. Average Book Price by Author (Only for Fiction Books)

Query Explanation:

This query calculates the average price of books for each author, **limited to the 'Fiction' genre**.



The screenshot shows the SQL Server Enterprise Manager interface. The Object Explorer on the left displays the database structure, including tables like authors, books, borrows, and members. The central pane shows a SQL query titled 'Coding Chall...Ragavi (73)' with the following code:

```
1 use librarydb;
2
3 select * from authors;
4 select * from books;
5 select * from borrows;
6 select * from members;
7
8 -- Average Book Price by Author (Only for Fiction books)
9
10 select a.name as author_name, avg(b.price) as avg_price
11 from books b
12 join authors a on b.author_id = a.author_id
13 where b.genre = 'Fiction'
14 group by a.name;
```

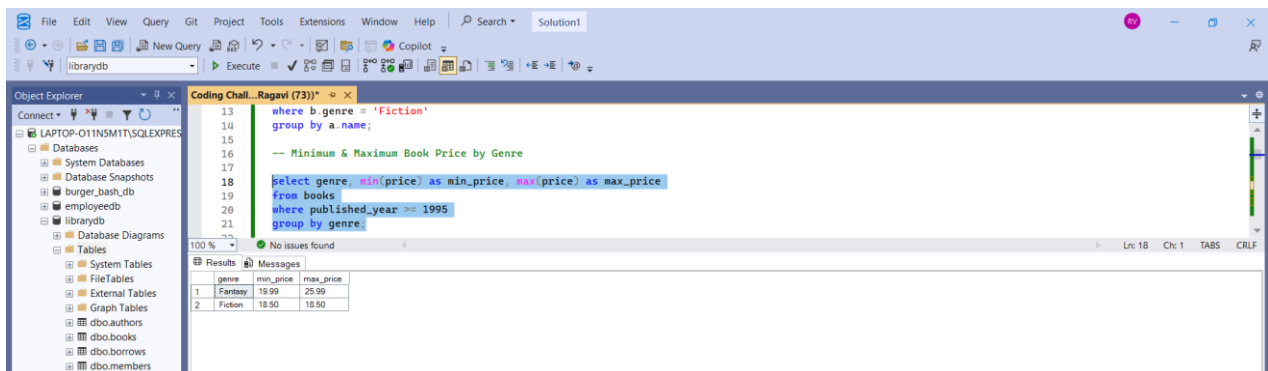
The Results pane at the bottom shows the output of the query:

author_name	avg_price
Haruki Murakami	18.500000

2. Minimum & Maximum Book Price by Genre (Published Year \geq 1995)

Query Explanation:

Finds the **minimum and maximum price** of books grouped by genre, but only includes books published **in or after 1995**.



The screenshot shows the SQL Server Enterprise Manager interface. The Object Explorer on the left displays the database structure. The central pane shows a SQL query titled 'Coding Chall...Ragavi (73)' with the following code:

```
13 where b.genre = 'Fiction'
14 group by a.name;
15
16 -- Minimum & Maximum Book Price by Genre
17
18 select genre, min(price) as min_price, max(price) as max_price
19 from books
20 where published_year >= 1995
21 group by genre;
```

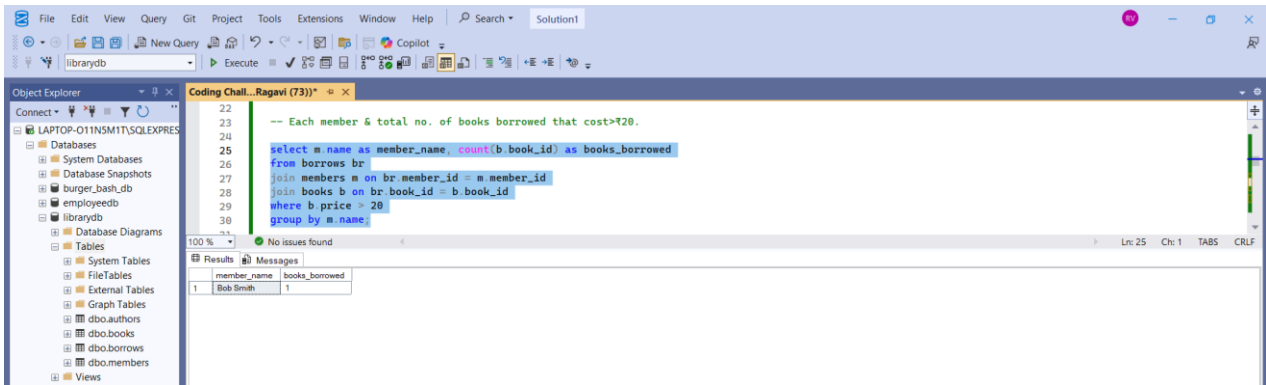
The Results pane at the bottom shows the output of the query:

genre	min_price	max_price
Fantasy	19.99	25.99
Fiction	18.50	18.50

3. Each Member & Total Books Borrowed That Cost $>$ ₹20

Query Explanation:

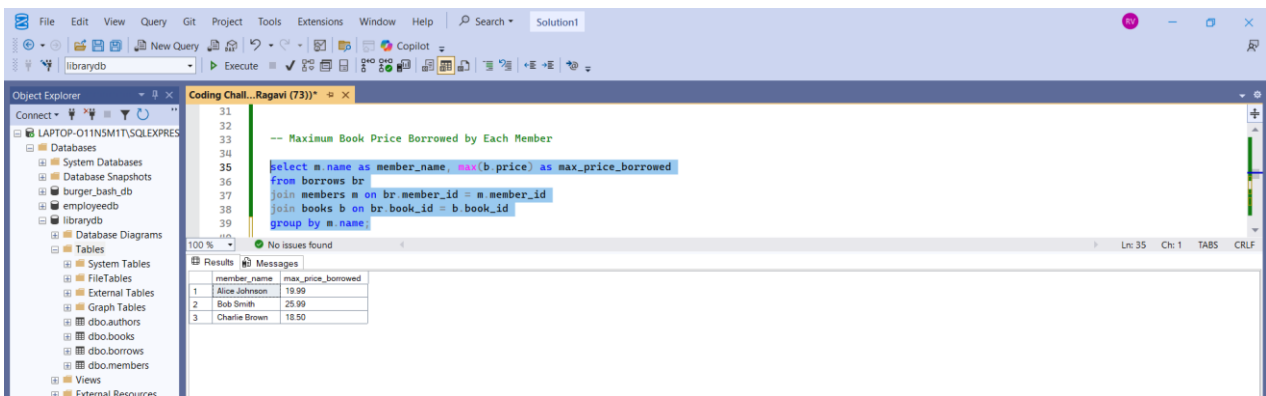
Returns each member's name along with the **count of books they borrowed** that cost **more than ₹20**.



4. Maximum Book Price Borrowed by Each Member

Query Explanation:

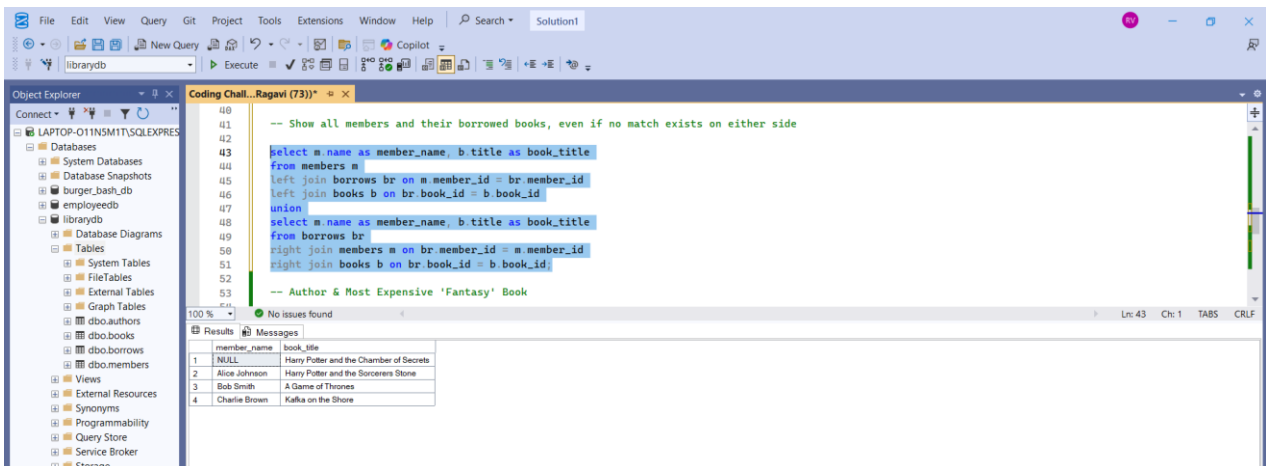
Shows the **highest-priced book** borrowed by each member.



5. Show All Members & Their Borrowed Books (Full Outer Join)

Query Explanation:

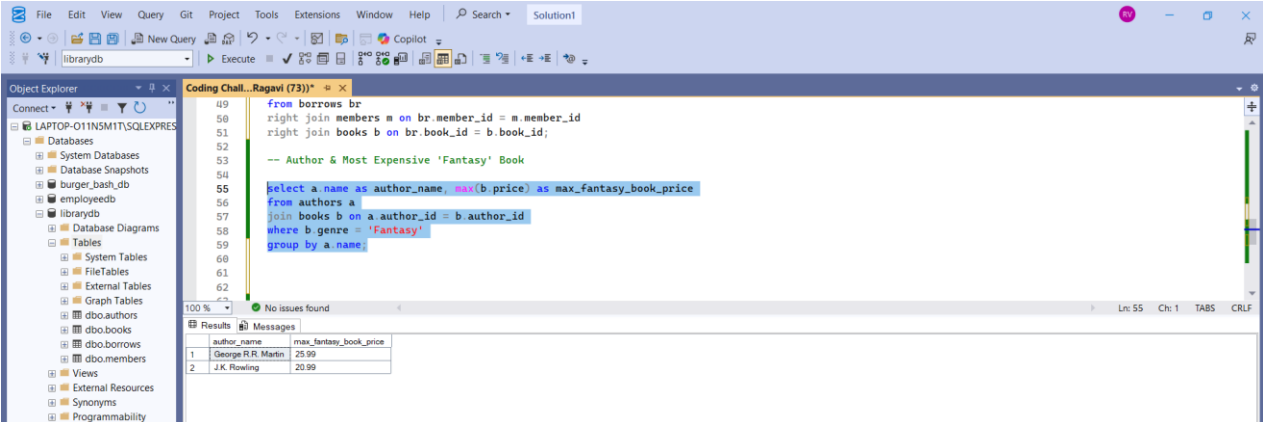
This query mimics a **FULL OUTER JOIN** using UNION. It lists all members and the books they borrowed, including those who haven't borrowed any books or books that haven't been borrowed yet.



6. Author & Most Expensive 'Fantasy' Book

Query Explanation:

Finds each author's name and the **highest-priced book** they've written in the **'Fantasy'** genre.



The screenshot shows the SQL Server Enterprise Manager interface. The left pane displays the Object Explorer with the 'librarydb' database selected. The center pane shows a SQL query titled 'Coding Chall...Ragavi (73)'. The query is as follows:

```
49 from borrows br
50 right join members m on br.member_id = m.member_id
51 right join books b on br.book_id = b.book_id;
52
53 -- Author & Most Expensive 'Fantasy' Book
54
55 select a.name as author_name, max(b.price) as max_fantasy_book_price
56 from authors a
57 join books b on a.author_id = b.author_id
58 where b.genre = 'Fantasy'
59 group by a.name;
```

The bottom pane shows the 'Results' tab with the following data:

	author_name	max_fantasy_book_price
1	George R.R. Martin	25.99
2	J.K. Rowling	20.99