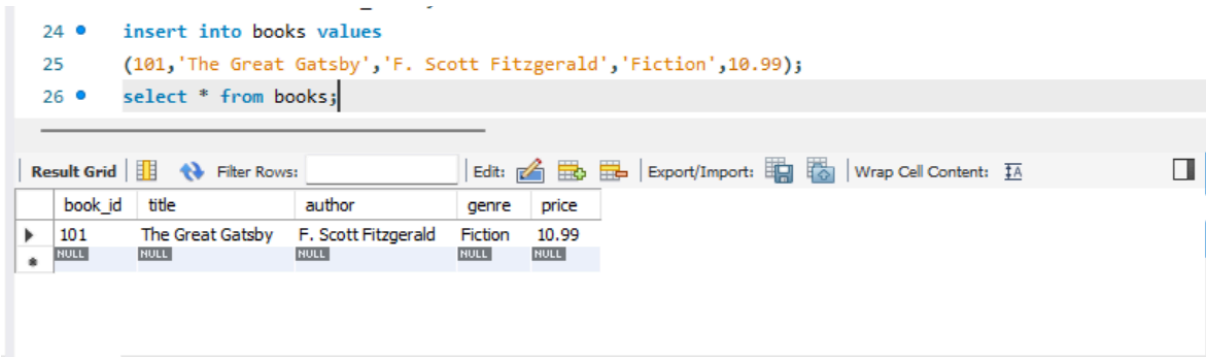
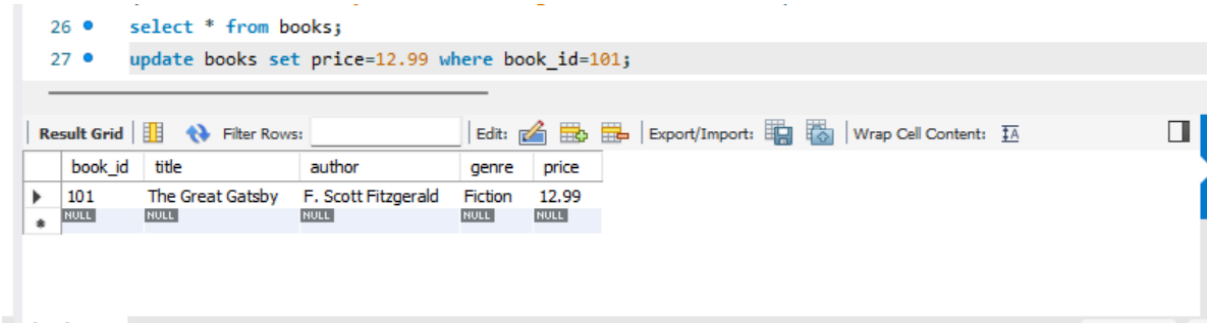
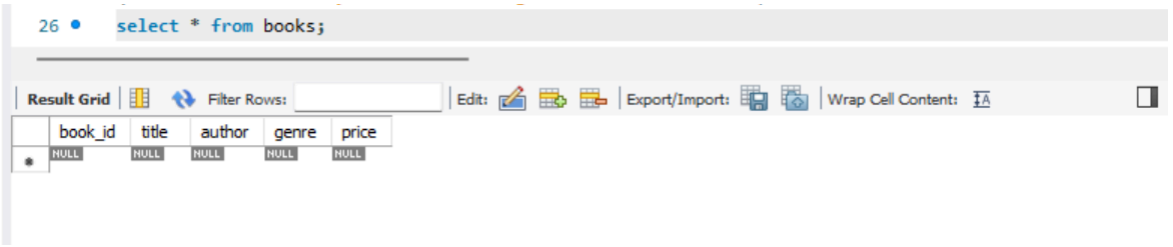
1. Insert a new book with the following details: book_id = 101, title = 'The Great Gatsby', author
= 'F. Scott Fitzgerald', genre = 'Fiction', price = 10.99.



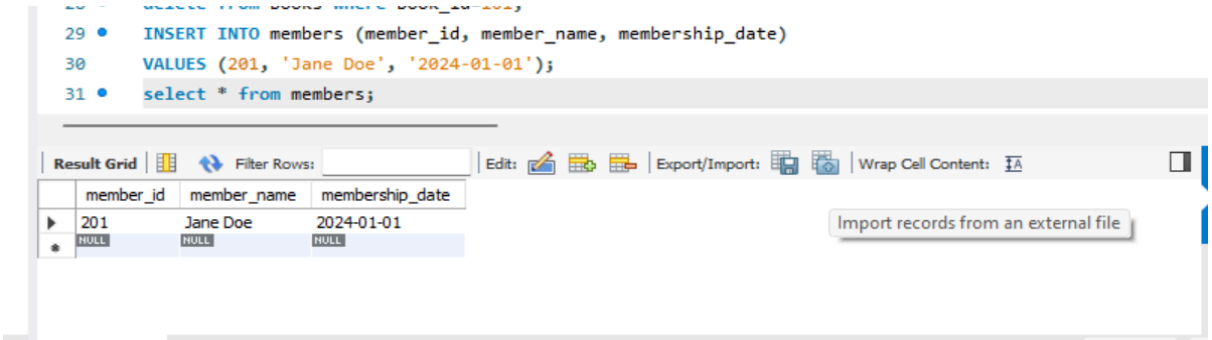
2. Update the price of the book with book_id = 101 to 12.99



3. Delete the book with book_id = 101



4. Insert a new member with the following details: member_id = 201, member_name = 'Jane Doe', membership_date = '2024-01-01'.



5. Update the membership_date of the member with member_id = 201 to '2024-02-01'.

```
32 •  UPDATE members
33    SET membership_date = '2024-02-01'
34    WHERE member_id = 201;
35
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

member_id	member_name	membership_date
201	Jane Doe	2024-02-01
NULL	NULL	NULL

6. Delete the member with member_id = 201.

```
35 •  DELETE FROM members
36    WHERE member_id = 201;
37
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

member_id	member_name	membership_date
NULL	NULL	NULL

7.Insert a new borrowed book record with the following details: borrow_id = 301, member_id = 202, book_id = 102, borrow_date = '2024-07-01', return_date = NULL.

```
37 •  INSERT INTO members (member_id, member_name, membership_date)
38    VALUES (202, 'abruti', '2024-07-09');
39 •  INSERT INTO books (book_id, title, author, genre, price)
40    VALUES (102, 'Book Title', 'Author Name', 'Genre', 10.99);
41 •  INSERT INTO borrowed_books (borrow_id, member_id, book_id, borrow_date, return_date)
42    VALUES (301, 202, 102, '2024-07-01', NULL);
43 •  select * from borrowed_books;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

borrow_id	member_id	book_id	borrow_date	return_date
301	202	102	2024-07-01	NULL
NULL	NULL	NULL	NULL	NULL

8. Update the return_date of the borrowed book record with borrow_id = 301 to '2024-07-15'.

```
43 •  select * from borrowed_books;
44 •  UPDATE borrowed_books
45    SET return_date = '2024-07-15'
46    WHERE borrow_id = 301;
47
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

borrow_id	member_id	book_id	borrow_date	return_date
301	202	102	2024-07-01	2024-07-15
NULL	NULL	NULL	NULL	NULL

9. Delete the borrowed book record with borrow_id = 301.

```
47 •  DELETE FROM borrowed_books
48    WHERE borrow_id = 301;
49 •  select * from borrowed_books;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

borrow_id	member_id	book_id	borrow_date	return_date
NULL	NULL	NULL	NULL	NULL

10. Count the total number of books.

```
50 •  SELECT COUNT(*) AS total_books
51    FROM books;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

total_books
1

11. Count the total number of members.

```
52 •  SELECT COUNT(*) AS total_members
53    FROM members;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

total_members
1

12. Count the total number of borrowed book records.

```
54 •  SELECT COUNT(*) AS total_borrowed_books
55    FROM borrowed_books;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

total_borrowed_books
0

13. Find the average price of books in the 'Fiction' genre.

```
56 •  SELECT AVG(price) AS average_price
57    FROM books
58    WHERE genre = 'Fiction';
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

average_price
NULL

14. Find the total number of books borrowed by each member.

```
59 •  SELECT member_id, COUNT(*) AS total_borrowed
60    FROM borrowed_books
61    GROUP BY member_id;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

member_id	total_borrowed

15. Find all books in the 'Fiction' genre.

```
62 •  SELECT *
63    FROM books
64    WHERE genre = 'Fiction';
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

book_id	title	author	genre	price
NULL	NULL	NULL	NULL	NULL

16. Find all members who joined after '2024-01-01'.

```
65 •  SELECT *
66    FROM members
67    WHERE membership_date > '2024-01-01';
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

member_id	member_name	membership_date
202	abruti	2024-07-09
NULL	NULL	NULL

17. Find all borrowed book records where the borrow_date is '2024-07-01'.

```
68 •  SELECT *
69    FROM borrowed_books
70    WHERE borrow_date = '2024-07-01';
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

borrow_id	member_id	book_id	borrow_date	return_date
NULL	NULL	NULL	NULL	NULL

18. Sort books by price in ascending order.

```
71 •  SELECT *
72    FROM books
73    ORDER BY price ASC;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

book_id	title	author	genre	price
102	Book Title	Author Name	Genre	10.99
NULL	NULL	NULL	NULL	NULL

19. Sort members by name in alphabetical order.

```
74 •  SELECT *
75    FROM members
76    ORDER BY member_name ASC;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

member_id	member_name	membership_date
202	abruti	2024-07-09
NULL	NULL	NULL

20. Find the top 5 most expensive books.

```
77 •  SELECT *
78    FROM books
79    ORDER BY price DESC
80    LIMIT 5;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

book_id	title	author	genre	price
102	Book Title	Author Name	Genre	10.99
NULL	NULL	NULL	NULL	NULL

21. Find members whose names start with 'J'.

```
81 •  SELECT *
82    FROM members
83    WHERE member_name LIKE 'J%';
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

member_id	member_name	membership_date
NULL	NULL	NULL

22. Find books with prices between 5 and 20.

```
84 •  SELECT *
85    FROM books
86    WHERE price BETWEEN 5 AND 20;
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

book_id	title	author	genre	price
102	Book Title	Author Name	Genre	10.99
NULL	NULL	NULL	NULL	NULL

23. Find members who have borrowed at least one book.

```
87 •  SELECT DISTINCT m.*
88    FROM members m
89    JOIN borrowed_books bb ON m.member_id = bb.member_id;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

member_id	member_name	membership_date

24. Find books that have not been borrowed yet.

```sql
SELECT *
FROM books
WHERE book_id NOT IN (SELECT DISTINCT book_id FROM borrowed_books);
```

Result Grid | Filter Rows: | Edit: | Export/Import: | Wrap Cell Content:

book_id	title	author	genre	price
102	Book Title	Author Name	Genre	10.99
NULL	NULL	NULL	NULL	NULL

25. Find the total amount spent on borrowed books by each member (considering book price).

```sql
SELECT m.member_id, m.member_name, SUM(b.price) AS total_spent
FROM members m
JOIN borrowed_books bb ON m.member_id = bb.member_id
JOIN books b ON bb.book_id = b.book_id
GROUP BY m.member_id, m.member_name;
```

Result Grid | Filter Rows: | Export: | Wrap Cell Content:

member_id	member_name	total_spent