

See /database folder for the SQL file with all the below queries. See Part 5 documentation for additional sample queries.

Query #1 (Fine Calculation): Calculate the total fines owed by each member, considering overdue books and a daily fine rate (e.g., \$0.25 per day).

```
SELECT
    c.ClientID,
    c.Name,
    SUM(
        CASE
            WHEN lt.ReturnDate IS NULL AND lt.DueDate < CURDATE() THEN
                DATEDIFF(CURDATE(), lt.DueDate) * 0.25
            WHEN lt.ReturnDate > lt.DueDate THEN DATEDIFF(lt.ReturnDate,
                lt.DueDate) * 0.25
            ELSE 0
        END
    ) AS TotalFine
FROM CLIENT c
JOIN LOAN_TRANSACTION lt ON c.ClientID = lt.ClientID
GROUP BY c.ClientID, c.Name;
```

```
MariaDB [library_management_system]> SELECT
-> c.ClientID,
-> c.Name,
-> SUM(
-> CASE
-> WHEN lt.ReturnDate IS NULL AND lt.DueDate < CURDATE() THEN DATEDIFF(CURDATE(), lt.DueDate) * 0.25
-> WHEN lt.ReturnDate > lt.DueDate THEN DATEDIFF(lt.ReturnDate, lt.DueDate) * 0.25
-> ELSE 0
-> END
-> ) AS TotalFine
-> FROM CLIENT c
-> JOIN LOAN_TRANSACTION lt ON c.ClientID = lt.ClientID
-> GROUP BY c.ClientID, c.Name;
```

ClientID	Name	TotalFine
1	John Smith	155.25
2	Emily Johnson	162.75
3	Michael Brown	164.75
4	Sarah Wilson	164.00
5	David Thompson	163.25
6	Jennifer Garcia	161.75
7	Robert Martinez	162.25
8	Lisa Anderson	160.75
9	James Taylor	162.75
10	Patricia Rodriguez	161.25
11	William Davis	0.00
12	Elizabeth Clark	0.25
13	Thomas White	0.00
14	Mary Harris	0.50
15	Christopher Lee	1.50
16	Jessica Young	1.50
17	Daniel Allen	1.25
18	Amanda Scott	2.25
19	Matthew King	2.00
20	Laura Nelson	1.50
21	Anthony Parker	2.00
22	Sophia Carter	1.75
23	Kevin Phillips	2.00
24	Rachel Evans	1.75
25	Joseph Turner	1.75

25 rows in set (0.005 sec)

Query #2 (Book Availability): Display a list of all available books (not currently borrowed) within a specific genre.

```
SELECT i.ItemID, i.Title, b.Author, b.Genre, i.AvailabilityStatus
FROM ITEM i
JOIN BOOK b ON i.ItemID = b.ItemID
WHERE i.AvailabilityStatus='available' AND b.Genre='Fiction';
```

```
MariaDB [library_management_system]> SELECT i.ItemID, i.Title, b.Author, b.Genre, i.AvailabilityStatus
-> FROM ITEM i
-> JOIN BOOK b ON i.ItemID = b.ItemID
-> WHERE i.AvailabilityStatus='available' AND b.Genre='Fiction';
```

ItemID	Title	Author	Genre	AvailabilityStatus
1	The Great Gatsby	F. Scott Fitzgerald	Fiction	available

1 row in set (0.001 sec)

Query #3 (Frequent Borrowers based on Genre): Identify the members who have borrowed the most books in a particular genre (e.g., "Mystery") in the last year.

```
SELECT c.ClientID, c.Name, COUNT(t.BorrowDate) AS Total_Borrowed, b.Genre
FROM CLIENT c
```

```

JOIN LOAN_TRANSACTION t on c.ClientID = t.ClientID
JOIN BOOK b on t.ItemID = b.ItemID
WHERE b.Genre='Fiction'
GROUP BY c.ClientID;

MariaDB [library_management_system]> SELECT c.ClientID, c.Name, COUNT(t.BorrowDate) AS Total_Borrowed, b.Genre
-> FROM CLIENT c
-> JOIN LOAN_TRANSACTION t on c.ClientID = t.ClientID
-> JOIN BOOK b on t.ItemID = b.ItemID
-> WHERE b.Genre='Fiction'
-> GROUP BY c.ClientID;
+-----+-----+-----+-----+
| ClientID | Name           | Total_Borrowed | Genre |
+-----+-----+-----+-----+
| 1        | John Smith    | 1              | Fiction |
| 4        | Sarah Wilson  | 1              | Fiction |
| 6        | Jennifer Garcia | 1              | Fiction |
| 11       | William Davis | 1              | Fiction |
| 16       | Jessica Young | 1              | Fiction |
+-----+-----+-----+-----+
5 rows in set (0.001 sec)

```

Query #4 (Books due Soon): Generate a report of all books due within the next week, sorted by due date.

```

SELECT lt.TransactionID, c.Name AS BorrowerName, i.Title, lt.DueDate
FROM LOAN_TRANSACTION lt
JOIN CLIENT c ON lt.ClientID = c.ClientID
JOIN ITEM i ON lt.ItemID = i.ItemID
WHERE lt.ReturnDate IS NULL AND lt.DueDate BETWEEN CURDATE() AND CURDATE() +
INTERVAL 7 DAY
ORDER BY lt.DueDate ASC;

MariaDB [library_management_system]> SELECT
-> c.ClientID,
-> c.Name,
-> COUNT(*) AS BorrowCount
-> FROM LOAN_TRANSACTION lt
-> JOIN CLIENT c ON lt.ClientID = c.ClientID
-> JOIN BOOK b ON lt.ItemID = b.ItemID
-> WHERE b.Genre = 'Mystery'
-> AND lt.BorrowDate >= CURDATE() - INTERVAL 1 YEAR
-> GROUP BY c.ClientID, c.Name
-> ORDER BY BorrowCount DESC
-> LIMIT 5;
Empty set (0.008 sec)

```

Query #5 (Members with Overdue Books): List all members who currently have at least one overdue book, along with the titles of the overdue books.

```
SELECT c.ClientID, c.Name, i.Title, lt.DueDate
FROM CLIENT c
JOIN LOAN_TRANSACTION lt ON c.ClientID = lt.ClientID
JOIN ITEM i ON lt.ItemID = i.ItemID
WHERE lt.ReturnDate IS NULL AND lt.DueDate < CURDATE();
```

```
MariaDB [library_management_system]> SELECT c.ClientID, c.Name, i.Title, lt.DueDate
-> FROM CLIENT c
tID = lt.ClientID
JOIN ITEM i ON lt.ItemID = i.ItemID
WHERE lt.ReturnDate IS NULL AND lt.DueDate < CURDATE() -> JOIN LOAN_TRANSACTION lt ON c.ClientID = lt.ClientID
-> JOIN ITEM i ON lt.ItemID = i.ItemID
-> WHERE lt.ReturnDate IS NULL AND lt.DueDate < CURDATE();
```

ClientID	Name	Title	DueDate
5	David Thompson	The Godfather	2023-07-22
2	Emily Johnson	The Social Network	2023-07-24
9	James Taylor	National Geographic	2023-07-24
6	Jennifer Garcia	The Catcher in the Rye	2023-07-28
8	Lisa Anderson	A Brief History of Time	2023-08-01
3	Michael Brown	Time	2023-07-27
3	Michael Brown	The Hobbit	2025-04-24
10	Patricia Rodriguez	Harry Potter and the Philosopher's Stone (Audiobook)	2023-07-30
7	Robert Martinez	Breaking Bad: Complete Series	2023-07-26
4	Sarah Wilson	The Lord of the Rings	2023-07-19

10 rows in set (0.001 sec)

Query #6 (Average Borrowing Time): Calculate the average number of days members borrow books for a specific genre.

```
SELECT b.Genre, AVG(DATEDIFF(lt.ReturnDate, lt.BorrowDate)) AS AverageBorrowDays
FROM LOAN_TRANSACTION lt
JOIN BOOK b ON lt.ItemID = b.ItemID
WHERE lt.ReturnDate IS NOT NULL AND b.Genre = 'Fiction'
GROUP BY b.Genre;
```

```
MariaDB [library_management_system]> SELECT b.Genre, AVG(DATEDIFF(lt.ReturnDate, lt.BorrowDate)) AS AverageBorrowDays
ACTION lt
JOIN B -> FROM LOAN_TRANSACTION lt
-> JOIN BOOK b ON lt.ItemID = b.ItemID
-> WHERE lt.ReturnDate IS NOT NULL AND b.Genre = 'Fiction'
-> GROUP BY b.Genre;
```

Genre	AverageBorrowDays
Fiction	170.0000

1 row in set (0.003 sec)

Query #7 (Most Popular Author in the Last Month): Determine the author whose books have been borrowed the most in the last month.

```
SELECT b.Author, COUNT(*) AS BorrowCount
FROM LOAN_TRANSACTION lt
JOIN BOOK b ON lt.ItemID = b.ItemID
WHERE lt.BorrowDate >= CURDATE() - INTERVAL 1 MONTH
GROUP BY b.Author
ORDER BY BorrowCount DESC
LIMIT 1;
```

```
MariaDB [library_management_system]> SELECT b.Author, COUNT(*) AS BorrowCount
-> FROM LOAN_TRANSACTION lt
-> JOIN BOOK b ON lt.ItemID = b.ItemID
-> WHERE lt.BorrowDate >= CURDATE() - INTERVAL 1 MONTH
-> GROUP BY b.Author
-> ORDER BY BorrowCount DESC
-> LIMIT 1;
```

Author	BorrowCount
J.R.R. Tolkien	1

1 row in set (0.014 sec)