Hackathon 1st Library Management System

```
-- Books Table
CREATE TABLE Books (
    BookID INT PRIMARY KEY AUTO_INCREMENT,
    ISBN VARCHAR(20) UNIQUE NOT NULL,
    Title VARCHAR(100) NOT NULL,
    Author VARCHAR(100) NOT NULL,
    Genre VARCHAR(50),
    Availability BOOLEAN DEFAULT TRUE
);
-- Users Table
CREATE TABLE Users (
    UserID INT PRIMARY KEY AUTO_INCREMENT,
    Name VARCHAR(100) NOT NULL,
    Email VARCHAR(100) UNIQUE NOT NULL
);
-- Transactions Table
CREATE TABLE Transactions (
    TransactionID INT PRIMARY KEY AUTO_INCREMENT,
    UserID INT,
    BookID INT,
    IssueDate DATE,
    ReturnDate DATE,
    FOREIGN KEY (UserID) REFERENCES Users(UserID),
    FOREIGN KEY (BookID) REFERENCES Books(BookID)
);
```

// sample datas

```
INSERT INTO Books (ISBN, Title, Author, Genre, Availability)
VALUES
('978-3-16-148410-0', 'The Alchemist', 'Paulo Coelho', 'Fiction', TRUE),
('978-0-7432-7356-5', 'Angels and Demons', 'Dan Brown', 'Thriller', TRUE);

INSERT INTO Users (Name, Email)
VALUES
('Arun Kumar', 'arun@example.com'),
('Priya Ramesh', 'priya@example.com'),
('Karthik Raja', 'karthik@example.com'),
('Deepa Suresh', 'deepa@example.com'),
('Vikram Anand', 'vikram@example.com'),
('Anjali Devi', 'anjali@example.com'),
('Ravi Chandran', 'ravi@example.com'),
('Meera Venkat', 'meera@example.com'),
('Sundar Mohan', 'sundar@example.com'),
('Lakshmi Narayan', 'lakshmi@example.com');

INSERT INTO Transactions (UserID, BookID, IssueDate, ReturnDate)
VALUES
(1, 1, '2025-03-01', '2025-03-10'),
(2, 2, '2025-03-05', NULL);
```

// Fetch Books by Genre

```
SELECT * FROM Books WHERE Genre = 'Fiction';
```

// Fetch Available Books

```
SELECT * FROM Books WHERE Availability = TRUE;
```

// Track Overdue Books

```
SELECT t.TransactionID, u.Name, b.Title, t.IssueDate, t.ReturnDate
FROM Transactions t
JOIN Users u ON t.UserID = u.UserID
JOIN Books b ON t.BookID = b.BookID
WHERE t.ReturnDate IS NULL AND t.IssueDate < CURDATE() - INTERVAL 14 DAY;
```

// User Borrowing History

```
SELECT u.Name, b.Title, t.IssueDate, t.ReturnDate
FROM Transactions t
JOIN Users u ON t.UserID = u.UserID
JOIN Books b ON t.BookID = b.BookID
WHERE u.UserID = 1;
```

// Generate Report on Late Returns

```
SELECT u.Name, b.Title, DATEDIFF(CURDATE(), t.IssueDate) AS DaysLate
FROM Transactions t
JOIN Users u ON t.UserID = u.UserID
JOIN Books b ON t.BookID = b.BookID
WHERE t.ReturnDate IS NULL AND DATEDIFF(CURDATE(), t.IssueDate) > 14;
```