Library Management System - DBMS Mini Project

Aim

To design and implement a Library Management System using SQL that stores books, authors, and borrower details with proper normalization up to 2NF, and uses triggers to automate data logging after a borrowing event.

Objectives

1. To normalize the database schema up to 2NF.
2. To create relational tables for books, authors, and borrowings.
3. To use primary and foreign keys to maintain relationships.
4. To implement a trigger that logs borrowing transactions automatically.

Description

The Library Management System (LMS) stores and manages details of books, authors, and borrowings. Initially, the schema was not normalized and contained data redundancies. We refined the schema up to the Second Normal Form (2NF) to separate repeating information into different tables. A trigger was also added to automatically record a log whenever a book is borrowed.

Concepts Used

- Relational Schema Design
- First and Second Normal Form (1NF, 2NF)
- SQL Commands (CREATE, INSERT, SELECT)
- Primary and Foreign Keys
- SQL Triggers

SQL Code

Author Table

```
CREATE TABLE Author (
    AuthorID INT PRIMARY KEY,
    AuthorName VARCHAR(50),
    AuthorCountry VARCHAR(50)
);
```

Book Table

```
CREATE TABLE Book (
    BookID INT PRIMARY KEY,
    BookTitle VARCHAR(100),
    AuthorID INT,
    FOREIGN KEY (AuthorID) REFERENCES Author(AuthorID)
);
```

Borrower Table

```
CREATE TABLE Borrower (
    BorrowerID INT PRIMARY KEY,
    BorrowerName VARCHAR(50)
);
```

Borrowing Table

```
CREATE TABLE Borrowing (
    BookID INT,
    BorrowerID INT,
    BorrowDate DATE,
    PRIMARY KEY (BookID, BorrowerID, BorrowDate),
    FOREIGN KEY (BookID) REFERENCES Book(BookID),
    FOREIGN KEY (BorrowerID) REFERENCES Borrower(BorrowerID)
);
```

Borrow Log Table

```
CREATE TABLE Borrow_Log (
    LogID INTEGER PRIMARY KEY AUTOINCREMENT,
    BookID INT,
    BorrowerID INT,
    BorrowDate DATE
);
```

Trigger

```
DELIMITER $$
CREATE TRIGGER log_borrowing
AFTER INSERT ON Borrowing
FOR EACH ROW
BEGIN
    INSERT INTO Borrow_Log (BookID, BorrowerID, BorrowDate)
    VALUES (NEW.BookID, NEW.BorrowerID, NEW.BorrowDate);
END$$
DELIMITER ;
```

Library Management System - DBMS Mini Project

Sample Data Insertion

```
INSERT INTO Author VALUES (1, 'J.K. Rowling', 'UK');
INSERT INTO Book VALUES (101, 'Harry Potter', 1);
INSERT INTO Borrower VALUES (201, 'Alice');
INSERT INTO Borrowing VALUES (101, 201, '2025-05-29');
```

Expected Output

```
SELECT * FROM Borrow_Log;

+-------+--------+-----------+------------+
| LogID | BookID | BorrowerID | BorrowDate  |
+-------+--------+-----------+------------+
| 1     |  101   |    201     | 2025-05-29  |
+-------+--------+-----------+------------+
```

Conclusion

This project shows how to implement a normalized Library Management System using SQL. It reduces redundancy and maintains data integrity using 2NF. Triggers automate logging when books are borrowed, making the system efficient and reliable. This mini project helps in understanding real-world database concepts.