

# Week 3 group Assessment

## Group Members :

1. Drishti Durgesh Telgu
2. Shubhreet Kaur

- **Library Management Database Design**

The Library Management Database is a systematically designed database for managing library resources and transactions. Thus, a relational database includes four central tables: Books, Authors, Members, and Loans.

The Books table will store information about each book, such as a unique BookID, title, genre, and the AuthorID corresponding with each book, thus establishing a direct relationship with the Authors table. The Authors table will contain complete information of authors, namely, the AuthorID, first name, last name, and a short biography, thereby allowing full comprehension of the different contributions made to the library collection by those authors.

The Members table is utilized for maintaining information about library members. It contains a unique MemberID, first and last name, and date of membership. In this way, we can keep track of who our active members are, and the amount of their usage with the library. There is also the Loans table, which simply logs the events of book loans. It contains a unique LoanID, BookID, MemberID, the date of loan, and return date. The table will be useful in the management of book

circulation so that information concerning the location of lent-out items is easily known.

Relationships among these tables include one-to-many between Authors and Books, and Members to Loans. This addition makes the database significantly functional as far as concern about tracking the availability of the books and the users' interaction is concerned. This structure supports not only operational efficiency but also enables robust reporting capabilities for library management toward informed decision-making in order to achieve better service delivery at the

libraries. Md Badiuzzaman (Bodi). (2024). *Library Management Database Design Assignment*.

Available at: <https://moodle.sydneyet.edu.au/mod/assign/view.php?id=12899> [Accessed 25 Oct. 2024].

- **Tables and their Fields**

1. Book Table

| Field Name | Data Type    | Description                        | Constraints                               |
|------------|--------------|------------------------------------|---|
| BookID     | Integer      | A unique identifier for each book. | Primary Key, Auto-increment               |
| Title      | Varchar(255) | The title of the book.             | Not Null                                  |
| AuthorID   | Integer      | References the author of the book. | Foreign Key (Authors.Author ID), Not Null |
| Genre      | Varchar(100) | The genre or category of the book. | Not Null                                  |

|               |         |                                  |                       |
|---------------|---------|----------------------------------|-----------------------|
| PublishedYear | Integer | The year the book was published. | Not Null, Must be > 0 |
|---------------|---------|----------------------------------|-----------------------|

## 2. Authors Table

| Field Name | Data Type    | Description                          | Constraints                 |
|------------|--------------|--------------------------------------|-----------------------------|
| AuthorID   | Integer      | A unique identifier for each author. | Primary Key, Auto-increment |
| FirstName  | Varchar(100) | The first name of the author.        | Not Null                    |
| LastName   | Varchar(100) | The last name of the author.         | Not Null                    |
| Bio        | Text         | A brief biography of the author.     | Optional                    |

## 3. Member Table

| Field Name | Data Type    | Description                          | Constraints                 |
|------------|--------------|--------------------------------------|-----------------------------|
| MemberID   | Integer      | A unique identifier for each member. | Primary Key, Auto-increment |
| FirstName  | Varchar(100) | The first name                       | Not Null                    |

|                |              |   |          |
|----------------|--------------|---|----------|
|                |              | of the member.                          |          |
| LastName       | Varchar(100) | The last name of the member.            | Not Null |
| MembershipDate | Date         | The date the member joined the library. | Not Null |

#### 4. Loan Table

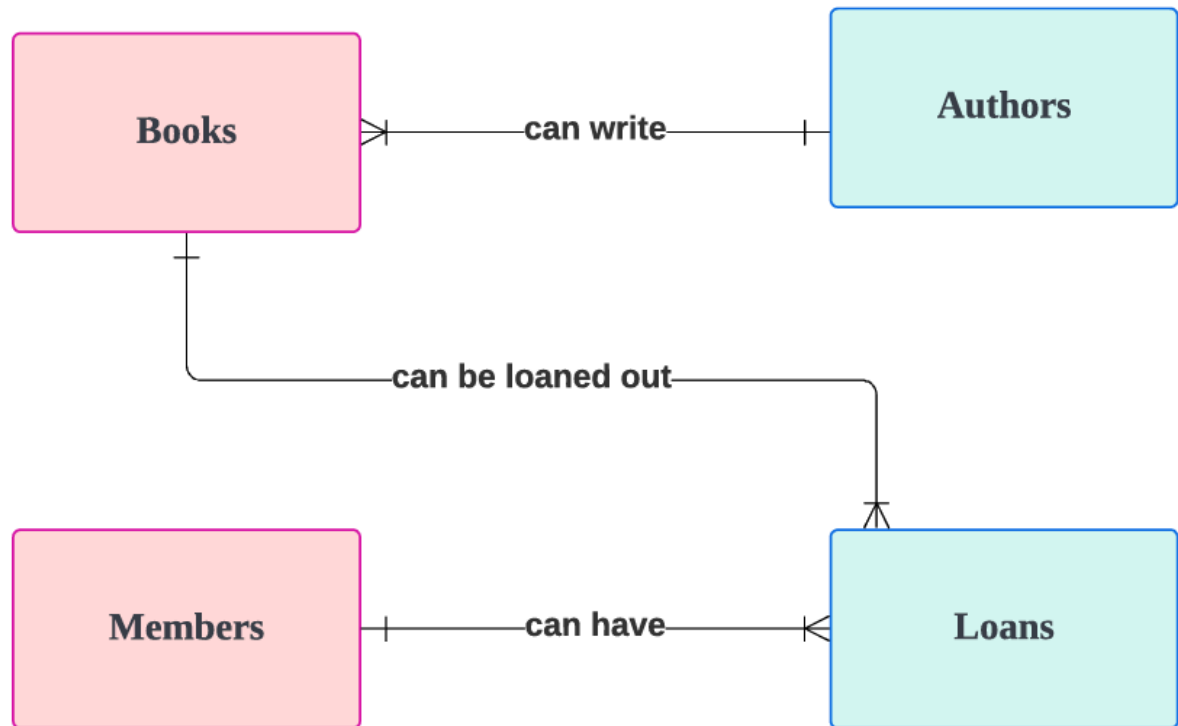
| Field Name | Data Type | Description                                    | Constraints                              |
|------------|-----------|--|--|
| LoanID     | Integer   | A unique identifier for each loan transaction. | Primary Key, Auto-increment              |
| BookID     | Integer   | References the book being loaned.              | Foreign Key (Books.BookID), Not Null     |
| MemberID   | Integer   | References the member who borrowed the book.   | Foreign Key (Members.MemberID), Not Null |
| LoanDate   | Date      | The date the book was                          | Not Null                                 |

|            |      |   |          |
|------------|------|---|----------|
|            |      | loaned out.   |          |
| ReturnDate | Date | The date the book was returned (can be NULL if not yet returned). | Optional |

- **Field Constraint :**

- **Primary Key:** Uniquely identifies each record in the table and cannot be null.
- **Foreign Key:** Creates a link between tables, ensuring referential integrity. It can only contain values that exist in the referenced table's primary key.
- **Not Null:** Indicates that a field must contain a value; it cannot be left empty.
- **Auto-increment:** Automatically generates a unique value for each new record.
- **Optional:** Indicates that a field can be left empty or null.

- **ER Diagram :**



Lucidchart. (n.d.). *Lucidchart: Diagram and flowchart software*. Available at: <https://www.lucidchart.com> [Accessed 25 Oct. 2024].

- **Conclusion :**

The ER diagram of the library management database effectively illustrates major relationships among four key entities involved, namely Books, Authors, Members, and Loans. One-to-many relationships show that the database is interrelated:

1. Books to Authors: Each author could also have several books assigned to him or her, thereby enabling a wide coverage of works an author has produced. This will enhance user experience in terms of ease of access and searching for an author's bibliography.
2. Members to Loans: The database is designed to capture the dynamic between library members and books borrowed. While a

member can take out many loans, this allows tracking of the history of borrowing and assists in effective management of library resources.

3. Books to Loans: While a book can be loaned many times, it pinpoints its popularity and demand among members. The relationship will be important to monitor the availability of the books and ensure efficient management of circulation.

In all, the designed ER diagram offers a robust framework for the Library Management Database, ensuring data integrity and relational dynamics. Such a structured approach not only supports the operational needs of the library but also enhances the generation of insightful reports and analyses, thus contributing to a more effective and user-friendly library system. Md Badiuzzaman (Bodi). (2024). *Library Management Database Design Assignment*. Available at: <https://moodle.sydney.net.edu.au/mod/assign/view.php?id=12899> [Accessed 25 Oct. 2024].

- **Reference :**

Lucidchart. (n.d.). *Lucidchart: Diagram and flowchart software*. Available at: <https://www.lucidchart.com> [Accessed 25 Oct. 2024].

Md Badiuzzaman (Bodi). (2024). *Library Management Database Design Assignment*. Available at: <https://moodle.sydney.net.edu.au/mod/assign/view.php?id=12899> [Accessed 25 Oct. 2024].