Library Management System

Efficiently manage library operations, books, and members.

Library Management System

The **Library Management System** project is a SQL-based system designed to streamline the management of books, members, and borrowing activities in a library. It keeps track of all the books available, the members who borrow them, and when they are borrowed or returned. The system ensures that library staff can easily manage book loans, monitor availability, and generate useful reports to see which books are currently borrowed, overdue, or available for lending.

In this project, I built the database structure, created tables for books, members, and borrowing records, and wrote SQL queries to manage and analyze the data. Some of the key tasks include listing books that are available or unavailable, identifying members who have borrowed but never returned books, and calculating the total number of borrowed books. I also created queries to find the member who borrowed a book for the longest time, track late returns, and display borrowing records for specific dates. This project demonstrates my ability to design a functional database and write effective SQL queries to retrieve useful information.

Structure of Books Table

Book ID

Unique identifier for each book.

Title

Name of the book.

Author

Author of the book.

Publication Year

Year of book publication.

Copies Available

No. Copies of Books that are Available.

Structure of Members Table

Member ID

Unique ID for each member.

Name

Full name of the member.

Email

Member's Email ID.

Phone Number

Member's contact number.

Structure of Record Table

Record ID

Unique ID for each record.

Book ID

Unique ID for each book.

Member ID

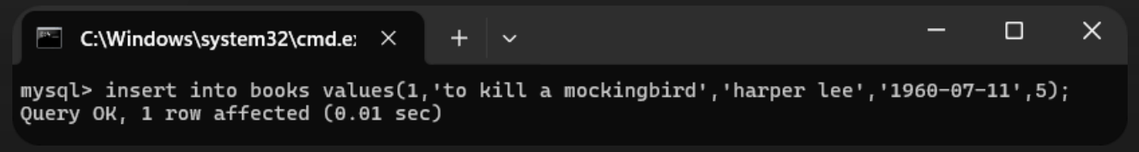
Unique ID for each member.

Borrow Date

Date of borrowing the book.

Return Date

Date of returning the book.



Adding a Book in the Library

Input Book Details

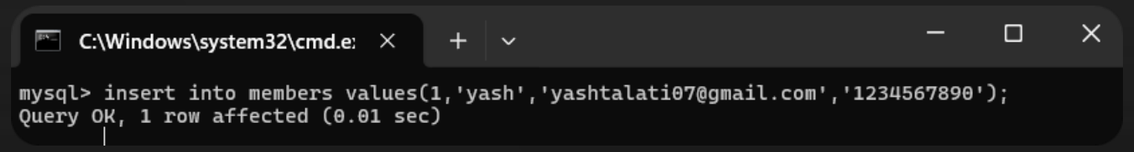
Gather information about the book: title, author, ISBN, genre, publication year, publisher, and language.

Validate and Store

Validate the entered data and store it in the Books table.

Update Inventory

Increment the book count in the library.



Adding a Member in the Library

Gather Member Info

Collect name, address, phone, email, and membership type.

Validate Input

Verify data accuracy and completeness.

Store in Members Table

Insert new member record into the Members table.

Borrowing a Book from the Library

```
C:\Windows\system32\cmd.e:

mysql> insert into borrowbooks values (1,1,1,'2024-08-23',NULL);
Query OK, 1 row affected (0.01 sec)

mysql> update books set copies_available = copies_available-1 where book_id=1;
Query OK, 1 row affected (0.01 sec)
Rows matched: 1  Changed: 1  Warnings: 0
```

Query

Record Table

Record added into the table Successfully.

```
C:\Windows\system32\cmd.e:

mysql> select * from borrowbooks;
+-----------+---------+-----------+------------+------------+
| record_id | book_id | member_id | borrowdate | returndate |
+-----------+---------+-----------+------------+------------+
|         1 |       1 |         1 | 2024-08-23 | NULL       |
+-----------+---------+-----------+------------+------------+
1 row in set (0.00 sec)
```

Books Table

Decrement in No. of Copy of Books Available.

```
C:\Windows\system32\cmd.e:

mysql> select * from books;
+---------+----------------------+-----------+---------------------+------------------+
| book_id | title                | author    | year_of_publication | copies_available |
+---------+----------------------+-----------+---------------------+------------------+
|       1 | to kill a mockingbird | harper lee | 1960-07-11          |                4 |
+---------+----------------------+-----------+---------------------+------------------+
1 row in set (0.00 sec)
```

Returning Book to the Library

```
C:\Windows\system32\cmd.e:

mysql> update borrowbooks set returndate = curdate() where record_id=1;
Query OK, 1 row affected (0.01 sec)
Rows matched: 1  Changed: 1  Warnings: 0

mysql> update books set copies_available = copies_available+1 where book_id=1;
Query OK, 1 row affected (0.01 sec)
Rows matched: 1  Changed: 1  Warnings: 0
```

Query

Record Table

Record table updated Successfully.

```
C:\Windows\system32\cmd.e:
mysql> select * from borrowbooks;
+-----------+---------+-----------+------------+------------+
| record_id | book_id | member_id | borrowdate | returndate |
+-----------+---------+-----------+------------+------------+
|         1 |       1 |         1 | 2024-08-23 | 2024-10-03 |
+-----------+---------+-----------+------------+------------+
1 row in set (0.00 sec)
```

Books Table

Increment in No. of Copy of Books Available.

```
C:\Windows\system32\cmd.e:
mysql> select * from books;
+---------+-----------------------+------------+---------------------+------------------+
| book_id | title                 | author     | year_of_publication | copies_available |
+---------+-----------------------+------------+---------------------+------------------+
|       1 | to kill a mockingbird | harper lee | 1960-07-11          |                5 |
+---------+-----------------------+------------+---------------------+------------------+
1 row in set (0.00 sec)
```

List Members Who Have never returned the books they borrowed

```
mysql> select m.name from members as m join borrowbooks as b on m.member_id = b.member_id where b.returndate IS NULL;
+--------------+
| name         |
+--------------+
| David Wilson |
| Eve Davis    |
| Frank Miller |
| Grace Taylor |
| Hannah Moore |
| Ivy Anderson |
| Jack Thomas  |
| Karen Lewis  |
| Liam Harris  |
| Mia Clark    |
+--------------+
10 rows in set (0.00 sec)
```

List all Available Books to Issue/Borrow

```
mysql> select title, author, year_of_publication, copies_available from books where copies_available >
 0;
+-------------------------+-------------------+---------------------+-------------------+
| title                   | author            | year_of_publication | copies_available |
+-------------------------+-------------------+---------------------+-------------------+
| to kill a mockingbird   | harper lee        | 1960-07-11          |                 3 |
| Pride and Prejudice     | Jane Austen       | 1813-01-28          |                10 |
| Moby Dick               | Herman Melville   | 1851-10-18          |                10 |
| The Catcher in the Rye  | J.D. Salinger     | 1951-07-16          |                 3 |
| Brave New World         | Aldous Huxley     | 1932-08-31          |                10 |
| Crime and Punishment    | Fyodor Dostoevsky | 1866-01-15          |                10 |
+-------------------------+-------------------+---------------------+-------------------+
6 rows in set (0.00 sec)
```

List all Unavailable Books

```
mysql> select title, author, year_of_publication, copies_available from books where copies_available >
 0;
+-------------------------+-------------------+---------------------+-------------------+
| title                   | author            | year_of_publication | copies_available |
+-------------------------+-------------------+---------------------+-------------------+
| to kill a mockingbird   | harper lee        | 1960-07-11          |                 3 |
| Pride and Prejudice     | Jane Austen       | 1813-01-28          |                10 |
| Moby Dick               | Herman Melville   | 1851-10-18          |                10 |
| The Catcher in the Rye  | J.D. Salinger     | 1951-07-16          |                 3 |
| Brave New World         | Aldous Huxley     | 1932-08-31          |                10 |
| Crime and Punishment    | Fyodor Dostoevsky | 1866-01-15          |                10 |
+-------------------------+-------------------+---------------------+-------------------+
6 rows in set (0.00 sec)
```

Find the Name of the Person and Time Period who have Issued the Book for the Longest Time from the Library

```
mysql> use libraryproject1;
Database changed
mysql> select m.name, max(datediff(b.returndate, b.borrowdate)) as maxdue
    -> from members as m
    -> join
    -> borrowbooks as b
    -> on
    -> m.member_id = b.member_id
    -> where
    -> b.returndate is not null
    -> group by m.name
    -> order by maxdue desc
    -> limit 1;
+------+--------+
| name | maxdue |
+------+--------+
| yash |     41 |
+------+--------+
1 row in set (0.01 sec)
```

Find Members Who Borrowed Books but Returned Them Late

```
mysql> select m.name from members as m
    -> join borrowbooks as b
    -> on m.member_id = b.member_id
    -> where datediff(b.returndate, b.borrowdate) > 15;
+------+
| name |
+------+
| yash |
+------+
1 row in set (0.01 sec)
```

Find Books Borrowed on a Specific Date

```
mysql> SELECT b.title, m.name
    -> FROM books as b
    -> JOIN borrowbooks as bb ON b.book_id = bb.book_id
    -> JOIN members as m ON bb.member_id = m.member_id
    -> WHERE bb.borrowdate = '2024-09-10';
+-----------------+--------------+
| title           | name         |
+-----------------+--------------+
| Moby Dick       | Bob Williams |
| Brave New World | Liam Harris  |
+-----------------+--------------+
2 rows in set (0.00 sec)
```

Calculate the Total Number of Books Borrowed

```
mysql> select count(record_id) as totalborrowedbooks from borrowbooks;
+--------------------+
| totalborrowedbooks |
+--------------------+
|                 33 |
+--------------------+
1 row in set (0.00 sec)
```

Calculate the Total Number of Available Books to Issue

```
mysql> select sum(copies_available) as booksavailable from books;
+----------------+
| booksavailable |
+----------------+
|             46 |
+----------------+
1 row in set (0.00 sec)
```

Find the Maximum and Minimum Borrowing Duration

```
mysql> select max(datediff(returndate, borrowdate)) as maxduration,
    -> min(datediff(returndate, borrowdate)) as minduration
    -> from borrowbooks where returndate is not null;
+-------------+-------------+
| maxduration | minduration |
+-------------+-------------+
|          41 |          14 |
+-------------+-------------+
1 row in set (0.00 sec)
```

Name the Members Who Have Borrowed Books from the Library

```
mysql> select m.name, count(bb.member_id) from
    -> members as m
    -> join borrowbooks as bb
    -> on m.member_id = bb.member_id
    -> group by m.name
    -> order by m.name;
+---------------+---------------------+
| name          | count(bb.member_id) |
+---------------+---------------------+
| Alice Johnson |                   2 |
| Bob Williams  |                   2 |
| Charlie Brown |                   2 |
| David Wilson  |                   2 |
| Eve Davis     |                   2 |
| Frank Miller  |                   2 |
| Grace Taylor  |                   2 |
| Hannah Moore  |                   2 |
| Ivy Anderson  |                   2 |
| Jack Thomas   |                   2 |
| Jane Smith    |                   2 |
| John Doe      |                   2 |
| Karen Lewis   |                   2 |
| Liam Harris   |                   2 |
| Mia Clark     |                   2 |
| yash          |                   3 |
+---------------+---------------------+
16 rows in set (0.01 sec)
```

Find the Book with the Highest Number of Copies

```
mysql> select title, copies_available
    -> from books
    -> order by copies_available desc
    -> limit 1;
+----------------------+------------------+
| title                | copies_available |
+----------------------+------------------+
| Crime and Punishment |               10 |
+----------------------+------------------+
1 row in set (0.00 sec)
```

Find the Total Number of Members Who Have Borrowed Books

```
mysql> SELECT COUNT(DISTINCT member_id) AS total_members_borrowed FROM borrowbooks;
+-----------------------+
| total_members_borrowed |
+-----------------------+
|                    16 |
+-----------------------+
1 row in set (0.00 sec)
```