Library Management SQL Project

This project demonstrates a Library Management System using SQL, focusing on queries involving joins, subqueries, REGEXP, UNION, GROUP BY, and more.

Features

- - Managing books and members.

- - Tracking borrowing records.

- - Advanced SQL queries for analysis.

```
Create database library;

use library;

CREATE TABLE Members (
    MemberID INT AUTO_INCREMENT PRIMARY KEY,
    Name VARCHAR(100),
    Age INT,
    City VARCHAR(50),
    Email VARCHAR(100)
);

INSERT INTO Members (Name, Age, City, Email) VALUES
('Alice Johnson', 25, 'New York', 'alice.johnson@example.com'),
('Bob Smith', 30, 'Los Angeles', 'bob.smith@example.com'),
('Charlie Brown', 22, 'Chicago', 'charlie.brown@example.com'),
('Diana Prince', 28, 'San Francisco', 'diana.prince@example.com'),
('Ethan Hunt', 35, 'Miami', 'ethan.hunt@example.com'),
('Fiona Gallagher', 24, 'Boston', 'fiona.gallagher@example.com'),
('George Wilson', 29, 'Seattle', 'george.wilson@example.com');
```

```
CREATE TABLE Books (
    BookID INT AUTO_INCREMENT PRIMARY KEY,
    Title VARCHAR(100),
    Author VARCHAR(100),
    Category VARCHAR(50),
    Price DECIMAL(8,2),
    PublishedYear YEAR,
    Available BOOLEAN
);

INSERT INTO Books (Title, Author, Category, Price, PublishedYear, Available) VALUES
('The Great Gatsby', 'F. Scott Fitzgerald', 'Fiction', 300.00, 1925, TRUE),
('Sapiens', 'Yuval Noah Harari', 'History', 450.00, 2011, TRUE),
('Atomic Habits', 'James Clear', 'Self-help', 400.00, 2018, TRUE),
('1984', 'George Orwell', 'Fiction', 250.00, 1949, FALSE),
('The Catcher in the Rye', 'J.D. Salinger', 'Fiction', 350.00, 1951, TRUE),
('Brief History of Time', 'Stephen Hawking', 'Science', 500.00, 1988, FALSE),
('Educated', 'Tara Westover', 'Biography', 450.00, 2018, TRUE),
('Becoming', 'Michelle Obama', 'Biography', 550.00, 2018, TRUE);


CREATE TABLE Borrowing (
    BorrowID INT AUTO_INCREMENT PRIMARY KEY,
    MemberID INT,
    BookID INT,
    BorrowDate DATE,
```

```
    ReturnDate DATE,
    FOREIGN KEY (MemberID) REFERENCES Members(MemberID),
    FOREIGN KEY (BookID) REFERENCES Books(BookID)
);

INSERT INTO Borrowing (MemberID, BookID, BorrowDate, ReturnDate) VALUES
(1, 1, '2024-01-10', '2024-01-20'),
(2, 2, '2024-01-15', '2024-01-25'),
(3, 3, '2024-02-01', NULL),
(4, 4, '2024-02-10', '2024-02-20'),
(5, 5, '2024-02-15', NULL),
(6, 6, '2024-03-01', '2024-03-10'),
(7, 7, '2024-03-05', NULL),
(1, 8, '2024-03-10', NULL);

-- sql queries
1) Find the names of members and the titles of books they borrowed?

SELECT m.Name, b.Title                          #join concept
FROM Borrowing br
JOIN Members m ON br.MemberID = m.MemberID
JOIN Books b ON br.BookID = b.BookID;

2) Find the books that were borrowed most frequently?
SELECT BookID, COUNT(BorrowID) AS BorrowCount
FROM Borrowing
GROUP BY BookID
```

```
HAVING COUNT(BorrowID) = (
    SELECT MAX(BorrowCount)
    FROM (
        SELECT BookID, COUNT(BorrowID) AS BorrowCount
        FROM Borrowing
        GROUP BY BookID
    ) AS SubQuery
);

SELECT b.Title, br.BorrowCount                          #Subquery concept
FROM (
    SELECT BookID, COUNT(BorrowID) AS BorrowCount
    FROM Borrowing
    GROUP BY BookID
    HAVING COUNT(BorrowID) = (
        SELECT MAX(BorrowCount)
        FROM (
            SELECT BookID, COUNT(BorrowID) AS BorrowCount
            FROM Borrowing
            GROUP BY BookID
        ) AS SubQuery
    )
) AS br
JOIN Books b ON br.BookID = b.BookID;
```

3)list all book titles along with titles of books published before2000?

```
select title, publishedyear from books
where publishedyear < 2000;
```

```
SELECT Title FROM Books
UNION
SELECT Title FROM Books WHERE PublishedYear < 2000;  # Union Concept
```

4)Count the number of members borrowing more than 5 books.

```
SELECT MemberID, COUNT(BookID) AS BooksBorrowed
FROM Borrowing
GROUP BY MemberID
HAVING COUNT(BookID) > 5;
```

5)List the names of members who have borrowed books that were published before the year 2000.?

```
SELECT DISTINCT m.Name
FROM Members m
JOIN Borrowing br ON m.MemberID = br.MemberID
JOIN Books b ON br.BookID = b.BookID
WHERE b.PublishedYear < 2000;
```

6) Write a query to display all books with their title, author, and the number of available copies.?

```
select * from books;
```

7)Identify the top 5 most borrowed books in the library.

```
select B.bookid,B.title,B.author,count(Br.BorrowId) as Borrow_count from books B
join Borrowing br on B.Bookid = Br.Bookid
```

```
group by B.BookID, B.title, B.author
order by borrow_count desc
limit 5;
```

8)Find all the books whose titles contain the word "Science"

```
SELECT Title
FROM Books
WHERE Title REGEXP 'Time';
```

9)Find all the books published between the years 1990 and 2010?

```
SELECT Title, PublishedYear
FROM Books
WHERE PublishedYear BETWEEN 1990 AND 2010;

SELECT COUNT(*) AS BookCount
FROM Books
WHERE PublishedYear BETWEEN 1990 AND 2010;
```