```
CREATE TABLE Books (
 BookId Int PRIMARY KEY,
 Title Varchar(50) Not Null,
 Author Varchar(50) Not Null,
 Publication Varchar(50) Not Null,
 ISBN Varchar(50) Not Null,
 Quantity Int Not Null
);

CREATE TABLE Members (
 MemberId Int PRIMARY KEY,
 FirstName Varchar(50) Not Null,
 LastName Varchar(50) Not Null,
 Email Varchar(50) Not Null,
 Contact Varchar(10) Not Null
);

CREATE TABLE IssuedBooks (
 TransactionId Int PRIMARY KEY,
 BookId Int Not Null,
 MemberId Int Not Null,
 IssueDate Date Not Null,
 ReturnDate Date
);
```

```
CREATE TABLE FineTable (
 FineId Int PRIMARY KEY,
 TransactionId Int Not Null,
 FineAmount Decimal(5,2) Not Null,
 FinePaid Decimal(5,2) Not Null
);
Insert Into Books Values (1, 'The Great Gatsby', 'F. Scott Fitzgerald', 'Scribner', '978-0743273565', 5);
Insert Into Books Values (2, 'To Kill a Mockingbird', 'Harper Lee', 'J. B. Lippincott', '978-0446310789', 3);
Insert Into Books Values (3, 'Pride and Prejudice', 'Jane Austen', 'T. Egerton', '978-0141439518', 7);
Insert Into Books Values (4, '1984', 'George Orwell', 'Secker and Warburg', '978-0451524935', 2);
Insert Into Books Values (5, 'The Hobbit', 'J. R. R. Tolkien', 'Allen & Unwin', '978-0547928227', 6);
Insert Into Members Values (1, 'John', 'Doe', 'john.doe@example.com', '9876543210');
Insert Into Members Values (2, 'Jane', 'Smith', 'jane.smith@example.com', '1234567890');
Insert Into Members Values (3, 'Mike', 'Johnson', 'mike.johnson@example.com', '9876543211');
Insert Into Members Values (4, 'Emily', 'Brown', 'emily.brown@example.com', '1234567891');
Insert Into Members Values (5, 'David', 'Anderson', 'david.anderson@example.com', '9876543212');
Insert Into IssuedBooks Values (1, 1, 2, '2022-03-01', '2022-03-10');
Insert Into IssuedBooks Values (2, 3, 4, '2022-04-05', '2022-04-15');
Insert Into IssuedBooks Values (3, 5, 1, '2022-02-15', Null);
Insert Into IssuedBooks Values (4, 2, 3, '2022-01-10', '2022-01-20');
Insert Into IssuedBooks Values (5, 1, 5, '2022-05-01', '2022-05-11');
Insert Into FineTable Values (1, 1, 20.00, 0);
Insert Into FineTable Values (2, 2, 15.00, 5.0);
Insert Into FineTable Values (3, 3, 25.00, 25.00);
Insert Into FineTable Values (4, 4, 10.00, 0);
Insert Into FineTable Values (5, 5, 30.00, 15);
```

```
-- Query to retrieve all the information from Books Table

SELECT *

FROM Books;

-- Query to retrieve the names of all the members from the members table

SELECT FirstName, LastName

FROM Members;

-- Query to retrieve the book name, author name, quantity, and publication of all the books

SELECT Title, Author, Quantity, Publication

FROM Books;

-- Query to retrieve the member name, book name, and issue Date for all the books borrowed by the member with ID 1

SELECT Members.FirstName, Members.LastName, Books.Title, IssuedBooks.IssueDate

FROM Members

INNER JOIN IssuedBooks ON Members.MemberId = IssuedBooks.MemberId

INNER JOIN Books ON IssuedBooks.BookId = Books.BookId

WHERE Members.MemberId = 1;

-- Query to retrieve the member ID, member name, and email of all the members whose name starts with 'J'

SELECT MemberId, FirstName, LastName, Email

FROM Members

WHERE FirstName LIKE 'J%';

-- Query to retrieve the member name, book name, and fine amount paid for all the members who have paid their complete fine

SELECT Members.FirstName, Members.LastName, Books.Title, FineTable.FinePaid

FROM Members

INNER JOIN IssuedBooks ON Members.MemberId = IssuedBooks.MemberId

INNER JOIN Books ON IssuedBooks.BookId = Books.BookId

INNER JOIN FineTable ON IssuedBooks.TransactionId = FineTable.TransactionId

WHERE FineTable.FineAmount = FineTable.FinePaid;
```

```
-- Query to find the name, email, contact, book name, and issue Date of all the members who have Not returned any book yet

SELECT m.FirstName, m.LastName, m.Email, m.Contact, b.Title, ib.IssueDate

FROM Members m

LEFT JOIN IssuedBooks ib ON m.MemberId = ib.MemberId

LEFT JOIN Books b ON ib.BookId = b.BookId

WHERE ib.ReturnDate IS Null;

-- Query to find the top most borrowed book along with its total number of borrowings

SELECT b.Title, COUNT(*) AS BorrowCount

FROM IssuedBooks ib

JOIN Books b ON ib.BookId = b.BookId

GROUP BY b.Title

ORDER BY BorrowCount DESC

LIMIT 1;
```