Library Management System

1. Create Books Table:

```
CREATE TABLE Books (
BookID INT AUTO_INCREMENT PRIMARY KEY,
Title VARCHAR(200) NOT NULL,
PublicationYear INT,
ISBN VARCHAR(15) UNIQUE NOT NULL
);
```

2. Create Authors Table:

```
CREATE TABLE Authors (
    AuthorID INT AUTO_INCREMENT PRIMARY KEY,
    AuthorName VARCHAR(100) NOT NULL
);
```

3. Create Book-Authors Table:

```
CREATE TABLE BookAuthors (
    BookAuthorID INT AUTO_INCREMENT PRIMARY KEY,
    BookID INT,
    AuthorID INT,
    FOREIGN KEY (BookID) REFERENCES Books(BookID),
    FOREIGN KEY (AuthorID) REFERENCES Authors(AuthorID)
);
```

4. Create Genres Table:

```
CREATE TABLE Genres (
    GenreID INT AUTO_INCREMENT PRIMARY KEY,
    GenreName VARCHAR(100) NOT NULL UNIQUE
);
```

5. Create Genre-Books Table:

```
CREATE TABLE BookGenres (
    BookGenreID INT AUTO_INCREMENT PRIMARY KEY,
    BookID INT,
    GenreID INT,
    FOREIGN KEY (BookID) REFERENCES Books(BookID),
    FOREIGN KEY (GenreID) REFERENCES Genres(GenreID)
);
```

6. Create MembershipType Table:

```
CREATE TABLE MembershipTypes (
    MembershipTypeID INT AUTO_INCREMENT PRIMARY KEY,
    TypeName VARCHAR(50) NOT NULL,
    MaxBooksAllowed INT NOT NULL
);
```

7. Create Members Table:

```
CREATE TABLE Members (
    MemberID INT AUTO_INCREMENT PRIMARY KEY,
    Name VARCHAR(100) NOT NULL,
    Address VARCHAR(200),
    Email VARCHAR(100),
    Phone INT,
    MembershipTypeID INT,
    FOREIGN KEY (MembershipTypeID) REFERENCES MembershipTypes(MembershipTypeID)
);
```

8. Create Book-Copies Table:

```
CREATE TABLE BookCopies (
    BookCopyID INT AUTO_INCREMENT PRIMARY KEY,
    BookID INT,
    Status VARCHAR(50),
    ConditionStatus VARCHAR(50),
    FOREIGN KEY (BookID) REFERENCES Books(BookID)
);
```

9. **Create Inventory Table:**

```
CREATE TABLE Inventory (
   InventoryID INT AUTO_INCREMENT PRIMARY KEY,
   BookCopyID INT,
   AcquisitionDate DATE,
   LastMaintenanceDate DATE,
   FOREIGN KEY (BookCopyID) REFERENCES BookCopies(BookCopyID)
);
```

10. **Create Borrowing-Information Table:**

```
CREATE TABLE BorrowingInfo (
   BorrowingID INT AUTO_INCREMENT PRIMARY KEY,
   MemberID INT,
   BookCopyID INT,
   BorrowDate DATE,
   DueDate DATE,
   ReturnDate DATE,
   Fine DECIMAL(10, 2),
   FOREIGN KEY (MemberID) REFERENCES Members(MemberID),
   FOREIGN KEY (BookCopyID) REFERENCES BookCopies(BookCopyID)
);
```

Solution of the given Question

1. **Find the book that has been borrowed the most times**

```
SELECT
   b.Title,
   COUNT(br.BookCopyID) AS BorrowCount
FROM
   BorrowingInfo br
JOIN
   BookCopies bc ON br.BookCopyID = bc.BookCopyID
JOIN
   Books b ON bc.BookID = b.BookID
GROUP BY
   bc.BookID
ORDER BY
   BorrowCount DESC
LIMIT 1;
```

2. **Calculate the average number of books borrowed per member**

```
SELECT
   AVG(BorrowCount)
FROM (
   SELECT
      MemberID,
      COUNT(BorrowingID) AS BorrowCount
   FROM
      borrowinginfo
   GROUP BY
      MemberID
) AS Dummy;
```

3. **Retrieve the minimum number of days a book was borrowed for**

```
SELECT
   MIN(DATEDIFF(ReturnDate, BorrowDate)) AS MinBorrowDays
FROM
   BorrowingInfo
WHERE
   ReturnDate IS NOT NULL;
```

4. **Find the member who borrowed the most books in the last year**

```
SELECT
   m.Name,
   COUNT(br.BookCopyID) AS BooksBorrowed
FROM
   BorrowingInfo br
JOIN
   Members m ON br.MemberID = m.MemberID
WHERE
   br.BorrowDate >= DATE_SUB(CURDATE(), INTERVAL 1 YEAR)
GROUP BY
   br.MemberID, m.Name
ORDER BY
   BooksBorrowed DESC
LIMIT 1;
```

5. **List the top 5 most borrowed book genres:**

```
SELECT
   g.GenreName,
   COUNT(br.BookCopyID) AS BorrowCount
FROM
   BorrowingInfo br
JOIN
   BookCopies bc ON br.BookCopyID = bc.BookCopyID
JOIN
   Books b ON bc.BookID = b.BookID
JOIN
   BookGenres bg ON b.BookID = bg.BookID
JOIN
   Genres g ON bg.GenreID = g.GenreID
GROUP BY
   g.GenreID
ORDER BY
   BorrowCount DESC
LIMIT 5;
```