

```
Query 1  SQL File 1*  SQL File 2*  SQL File 3*  SQL File 6*  SQL File 7* x  SQL File 8*
Limit to 1000 rows

1 • CREATE DATABASE LibraryDB;
2
3 • USE LibraryDB;
4
5 • CREATE TABLE Books (
6     BookID INT PRIMARY KEY AUTO_INCREMENT,
7     Title VARCHAR(255) NOT NULL,
8     PublishYear INT,
9     ISBN VARCHAR(20) UNIQUE,
10    Author VARCHAR(255) NOT NULL
11 );
12
13 • CREATE TABLE Members (
14     MemberID INT PRIMARY KEY AUTO_INCREMENT,
15     Name VARCHAR(255) NOT NULL,
16     Address VARCHAR(255),
17     Email VARCHAR(100) UNIQUE,
18     MembershipYear INT,
19     PhoneNo VARCHAR(15)
20 );
21
22 • CREATE TABLE BorrowedBooks (
23     BorrowID INT PRIMARY KEY AUTO_INCREMENT,
24     BookID INT,
25     MemberID INT,
26     BorrowDate DATE,
27     ReturnDate DATE,
28     FOREIGN KEY (BookID) REFERENCES Books(BookID),
29     FOREIGN KEY (MemberID) REFERENCES Members(MemberID)
30 );
```

Database connection

```
Source  History
1  /*
2   * Click nbfs://nbhost/SystemFileSystem/Templates/Licenses/license-default.txt to change this license
3   * Click nbfs://nbhost/SystemFileSystem/Templates/Classes/Main.java to edit this template
4   */
5   package librarymanagementsys;
6
7   /**
8    *
9    * @author bbhuv
10   */
11  import java.sql.*;
12  import java.util.Scanner;
13
14  public class LibraryManagementsys {
15      static final String URL = "jdbc:mysql://localhost:3306/LibraryDB";
16      static final String USER = "root";
17      static final String PASSWORD = "bhuvan@123";
18
19      public static void main(String[] args) {
20          Scanner sc = new Scanner(System.in);
21          try (Connection con = DriverManager.getConnection(URL, USER, PASSWORD)) {
22              while (true) {
23                  System.out.println("\nLibrary Management System");
24                  System.out.println("1. Add Book");
25                  System.out.println("2. View Books");
26                  System.out.println("3. Add Member");
27                  System.out.println("4. View Members");
28                  System.out.println("5. Borrow Book");
29                  System.out.println("6. Return Book");
30                  System.out.println("7. Exit");
31                  System.out.print("Enter choice: ");
32                  int choice = sc.nextInt();
33                  sc.nextLine();
```

```

34
35
36         switch (choice) {
37             case 1:
38                 addBook(con, sc);
39                 break;
40             case 2:
41                 viewBooks(con);
42                 break;
43             case 3:
44                 addMember(con, sc);
45                 break;
46             case 4:
47                 viewMembers(con);
48                 break;
49             case 5:
50                 borrowBook(con, sc);
51                 break;
52             case 6:
53                 returnBook(con, sc);
54                 break;
55             case 7:
56                 System.out.println("Exiting...");
57                 return;
58             default:
59                 System.out.println("Invalid choice! Try again.");
60         }
61     } catch (Exception e) {
62         System.out.println("Database Error: " + e.getMessage());
63     }
64 }
65

```

```

66 private static void addBook(Connection con, Scanner sc) throws SQLException {
67     System.out.print("Enter title: ");
68     String title = sc.nextLine();
69     System.out.print("Enter publish year: ");
70     int year = sc.nextInt();
71     sc.nextLine();
72     System.out.print("Enter ISBN: ");
73     String isbn = sc.nextLine();
74     System.out.print("Enter author: ");
75     String author = sc.nextLine();
76
77     String query = "INSERT INTO Books (Title, PublishYear, ISBN, Author) VALUES (?, ?, ?, ?)";
78     try (PreparedStatement pst = con.prepareStatement(query)) {
79         pst.setString(1, title);
80         pst.setInt(2, year);
81         pst.setString(3, isbn);
82         pst.setString(4, author);
83         pst.executeUpdate();
84         System.out.println("Book added successfully!");
85     }
86 }
87

```

```

87
88 private static void viewBooks(Connection con) throws SQLException {
89     String query = "SELECT * FROM Books";
90     try (Statement stmt = con.createStatement(); ResultSet rs = stmt.executeQuery(query)) {
91         while (rs.next()) {
92             System.out.println("BookID: " + rs.getInt("BookID") + ", Title: " + rs.getString("Title") +
93                 ", Publish Year: " + rs.getInt("PublishYear") + ", ISBN: " + rs.getString("ISBN") +
94                 ", Author: " + rs.getString("Author"));
95         }
96     }
97 }

```

```

99 private static void addMember(Connection con, Scanner sc) throws SQLException {
100     System.out.print("Enter name: ");
101     String name = sc.nextLine();
102     System.out.print("Enter address: ");
103     String address = sc.nextLine();
104     System.out.print("Enter email: ");
105     String email = sc.nextLine();
106     System.out.print("Enter membership year: ");
107     int year = sc.nextInt();
108     sc.nextLine();
109     System.out.print("Enter phone number: ");
110     String phone = sc.nextLine();
111
112     String query = "INSERT INTO Members (Name, Address, Email, MembershipYear, PhoneNo) VALUES (?, ?, ?, ?, ?)";
113     try (PreparedStatement pst = con.prepareStatement(query)) {
114         pst.setString(1, name);
115         pst.setString(2, address);
116         pst.setString(3, email);
117         pst.setInt(4, year);
118         pst.setString(5, phone);
119         pst.executeUpdate();
120         System.out.println("Member added successfully!");
121     }
122 }
123
124 private static void viewMembers(Connection con) throws SQLException {
125     String query = "SELECT * FROM Members";
126     try (Statement stmt = con.createStatement(); ResultSet rs = stmt.executeQuery(query)) {
127         while (rs.next()) {
128             System.out.println("MemberID: " + rs.getInt("MemberID") + ", Name: " + rs.getString("Name") +
129                 ", Address: " + rs.getString("Address") + ", Email: " + rs.getString("Email") +
130                 ", Membership Year: " + rs.getInt("MembershipYear") + ", Phone No: " + rs.getString("PhoneNo"));
131         }
132     }
133 }
134
135 private static void borrowBook(Connection con, Scanner sc) throws SQLException {
136     System.out.print("Enter Book ID: ");
137     int bookID = sc.nextInt();
138     System.out.print("Enter Member ID: ");
139     int memberID = sc.nextInt();
140     sc.nextLine();
141
142     String query = "INSERT INTO BorrowedBooks (BookID, MemberID, BorrowDate) VALUES (?, ?, CURDATE())";
143     try (PreparedStatement pst = con.prepareStatement(query)) {
144         pst.setInt(1, bookID);
145         pst.setInt(2, memberID);
146         pst.executeUpdate();
147         System.out.println("Book borrowed successfully!");
148     }
149 }
150
151 private static void returnBook(Connection con, Scanner sc) throws SQLException {
152     System.out.print("Enter Borrow ID: ");
153     int borrowID = sc.nextInt();
154     sc.nextLine();
155
156     String query = "UPDATE BorrowedBooks SET ReturnDate = CURDATE() WHERE BorrowID = ?";
157     try (PreparedStatement pst = con.prepareStatement(query)) {
158         pst.setInt(1, borrowID);
159         pst.executeUpdate();
160         System.out.println("Book returned successfully!");
161     }
162 }
163 }

```