@Entity

public class User {

@Id

@GeneratedValue(strategy = GenerationType.IDENTITY)

private Long id;

private String username;

private String password;

private String email;

private String fullName;

@Enumerated(EnumType.STRING)

private Role role;

}

public enum Role {

ADMIN,

LIBRARIAN,

MEMBER

}

@Entity

public class Book {

@Id

@GeneratedValue(strategy = GenerationType.IDENTITY)

private Long id;

private String title;

private String author;

private String isbn;

private String publisher;

private String genre;

private int copiesAvailable;

// Getters and setters

}

@Entity

public class Member {

@Id

@GeneratedValue(strategy = GenerationType.IDENTITY)

private Long id;

private String name;

private String email;

private String address;

private String phoneNumber;

// One-to-one relationship with User

@OneToOne

private User user;

// Getters and setters

}

@Entity

public class BorrowingRecord {

@Id

@GeneratedValue(strategy = GenerationType.IDENTITY)

private Long id;

@ManyToOne

private Member member;

@ManyToOne

private Book book;

private LocalDate borrowDate;

private LocalDate dueDate;

private LocalDate returnDate;

private BigDecimal fineAmount;

// Getters and setters

}

@Entity

public class Reservation {

@Id

@GeneratedValue(strategy = GenerationType.IDENTITY)

private Long id;

@ManyToOne

private Member member;

@ManyToOne

private Book book;

private LocalDate reservationDate;

private LocalDate notificationDate;

private boolean isActive;

// Getters and setters

}