

Library Management System using Hibernate

Hibernate Configuration

```
<hibernate-configuration>
  <session-factory>
    <property name="hibernate.connection.driver_class">com.mysql.cj.jdbc.Driver</property>
    <property name="hibernate.connection.url">jdbc:mysql://localhost:3306/librarydb</property>
    <property name="hibernate.connection.username">root</property>
    <property name="hibernate.connection.password">password</property>
    <property name="hibernate.dialect">org.hibernate.dialect.MySQL8Dialect</property>
    <property name="show_sql">true</property>
    <property name="hbm2ddl.auto">update</property>
    <mapping class="com.example.library.model.Book"/>
    <mapping class="com.example.library.model.Member"/>
    <mapping class="com.example.library.model.LibraryTransaction"/>
  </session-factory>
</hibernate-configuration>
```

Entity Classes

model/Book.java

```
@Entity
@Table(name = "books")
public class Book {
    @Id @GeneratedValue(strategy = GenerationType.IDENTITY)
    private int id;

    private String title;
    private String author;
    private int quantity;

    // getters & setters
}
```

model/Member.java

```
@Entity
@Table(name = "members")
public class Member {
    @Id @GeneratedValue(strategy = GenerationType.IDENTITY)
    private int id;

    private String name;
    private String email;

    // getters & setters
}
```

model/LibraryTransaction.java

```
@Entity
@Table(name = "library_transactions")
public class LibraryTransaction {
```

```

@Id @GeneratedValue(strategy = GenerationType.IDENTITY)
private int id;

private String type;        // ISSUE or RETURN
@Temporal(TemporalType.TIMESTAMP)
private Date date;

@ManyToOne private Book book;
@ManyToOne private Member member;

// getters & setters
}

```

Main Application Class

```

public class Main {
    public static void main(String[] args) {
        LibraryService service = new LibraryService();

        // create a member
        Member mem = new Member();
        mem.setName("John Doe");
        mem.setEmail("john@example.com");
        service.registerMember(mem);

        // create a book
        Book book = new Book();
        book.setTitle("Hibernate in Action");
        book.setAuthor("Gavin King");
        book.setQuantity(10);
        service.addBook(book);

        // issue & return
        service.issueBook(book.getId(), mem.getId());
        service.returnBook(book.getId(), mem.getId());

        HibernateUtil.getSessionFactory().close(); // tidy shutdown
    }
}

```

Sample Output

```

Member 'John Doe' registered.
Book 'Hibernate in Action' added.
Book issued to John Doe.
Book returned by John Doe.

```