Case Study
On
Used book selling website

INTRODUCTION

- The Used Book Selling Web Application is designed to provide an online platform for buying and selling second-hand books.
  The project aims to reduce the cost of educational and non-academic books by encouraging reuse.
  It allows sellers to list used books and buyers to browse available books easily.
  The application simplifies the process of book exchange by directly connecting buyers and sellers.
  It also helps reduce wastage and promotes sustainable learning practices.

ABSTRAT

- The Used Book Selling Web Application is a full-stack system developed using Java, Spring Boot, MySQL, HTML, CSS, and Thymeleaf.
  The system supports CRUD operations for managing book listings.
  A secure login system allows only authenticated sellers to add or delete books.
  Buyers can directly contact sellers using WhatsApp integration.
  The application follows the MVC architecture to ensure scalability and maintainability.

What we are going to build: Client Requirement

- The client requires a system where sellers can sell used books online.
- Buyers should be able to view available books without logging in.
- Sellers must be able to register, login, and manage their book listings.
- Each book should display seller contact details.
- The system should support secure data storage and easy navigation.

What we are going to build: Some technical terms.

- Seller should be able to register and login
- Seller should be able to add, update, and delete books
- Buyer should be able to view all available books
- Buyer should be able to contact seller via WhatsApp
- System should display book status (Available / Sold)

What are the technologies and tools we are going use ?

- Backend: Java, Spring Boot
- Frontend: HTML, CSS, Bootstrap, Thymeleaf
- Database: MySQL
- ORM: Spring Data JPA (Hibernate)
- IDE: Spring Tool Suite (STS)
- Web Server: Apache Tomcat
- Browser: Google Chrome

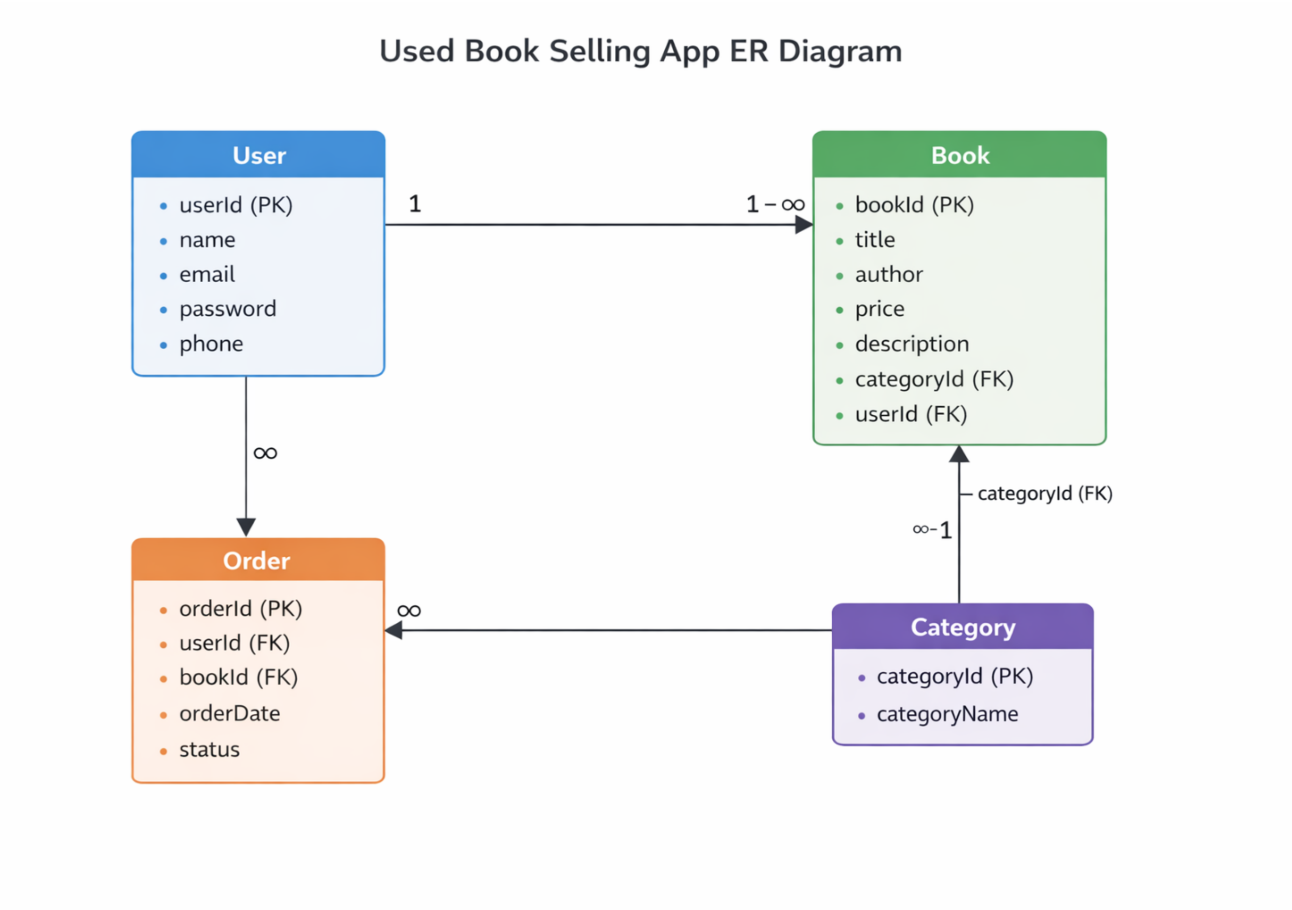
SYSTEM REQUIREMENTS

- **Software Requirements**
- • Operating System: Windows / Linux
  • IDE: Spring Tool Suite (STS)
  • Database Server: MySQL 8.0
  • Web Server: Apache Tomcat
  • Browser: Google Chrome

PROJECT MODULE

- Seller Module
  - Book Module
  - Authentication Module
  - Buyer Interaction Module

ER DIAGRAM



Used Book Selling App ER Diagram

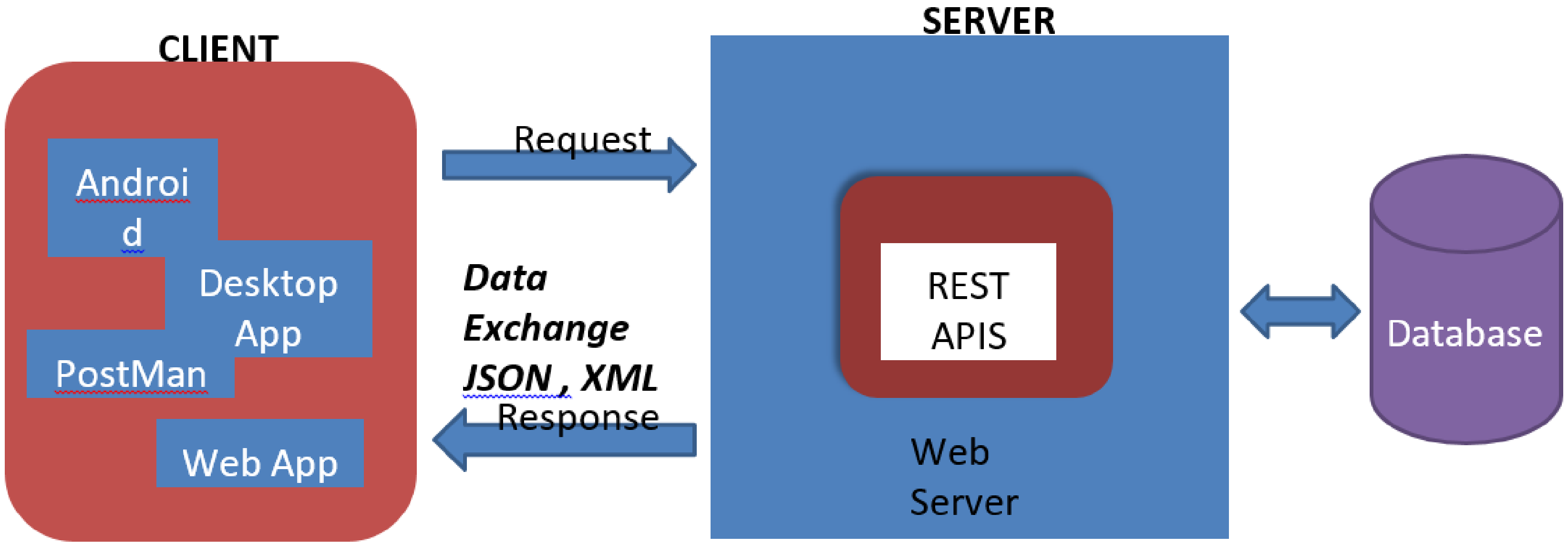
User
- userId (PK)
- name
- email
- password
- phone

Book
- bookId (PK)
- title
- author
- price
- description
- categoryId (FK)
- userId (FK)

Order
- orderId (PK)
- userId (FK)
- bookId (FK)
- orderDate
- status

Category
- categoryId (PK)
- categoryName

Client – Server Architecture



CLIENT

Android

Desktop App

PostMan

Web App

Request

Data Exchange JSON , XML

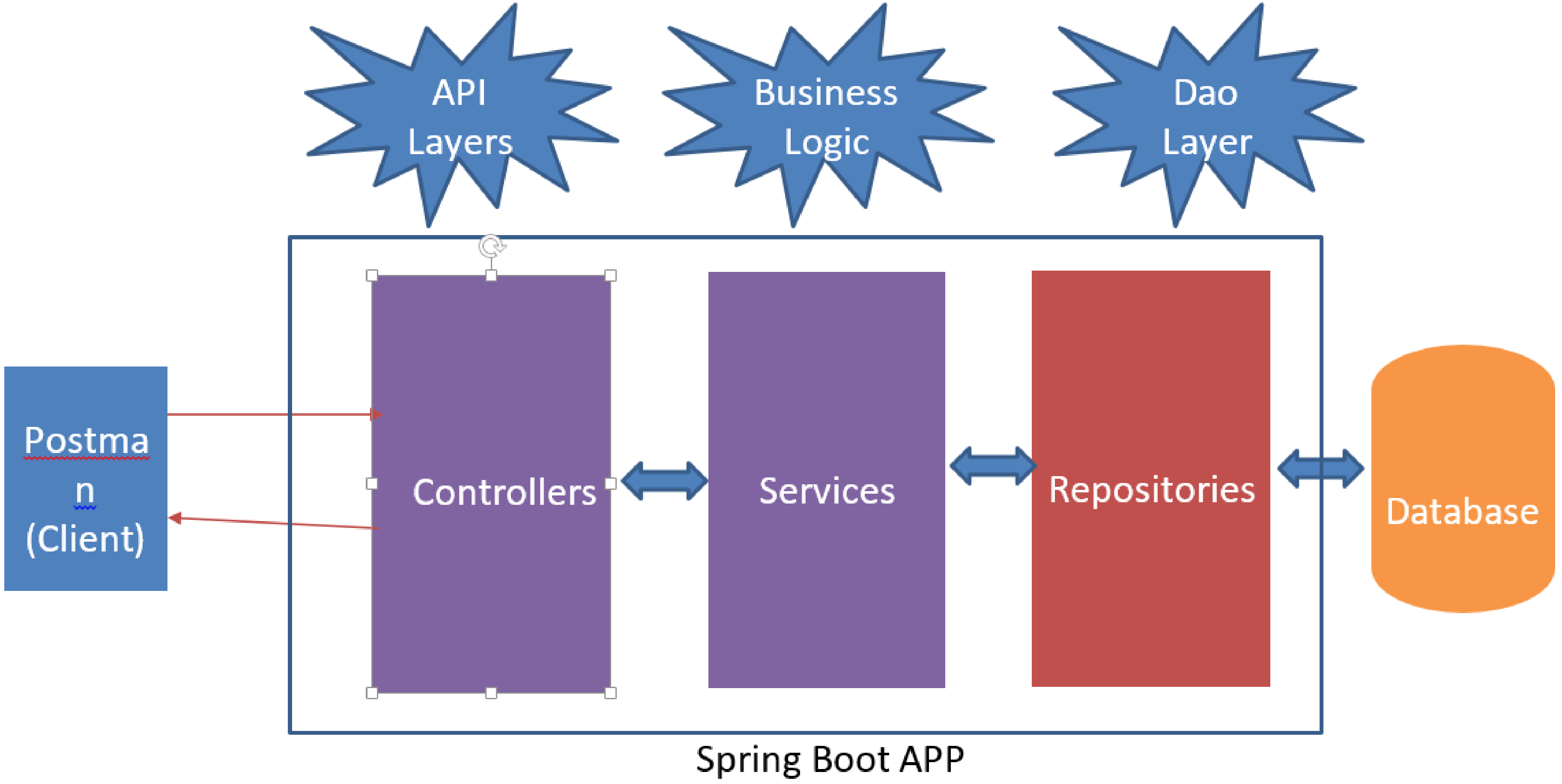
Response

SERVER

REST APIS

Web Server

Database


API Layers
Business Logic
Dao Layer
Postman (Client)
Controllers
Services
Repositories
Database
Spring Boot APP

SELLER MODULE

- The Seller Module allows users to register and login into the system.
- Only authenticated sellers can add or delete book listings.
- Seller details such as name and phone number are stored securely.
- Seller information is linked to the books they upload.

BOOK MODULE

- The Book Module manages all book-related operations.
- Sellers can add new books with title, author, price, and description.
- Books can be updated or deleted by the seller.
- All available books are displayed on the home page.

BUYER INTERACTION MODULE

- Buyers can view all available books without logging in.
- Each book listing displays seller contact details.
- Buyers can contact sellers directly using WhatsApp integration.
- This ensures fast and direct communication between users.

HTTP REQUEST METHODS USED

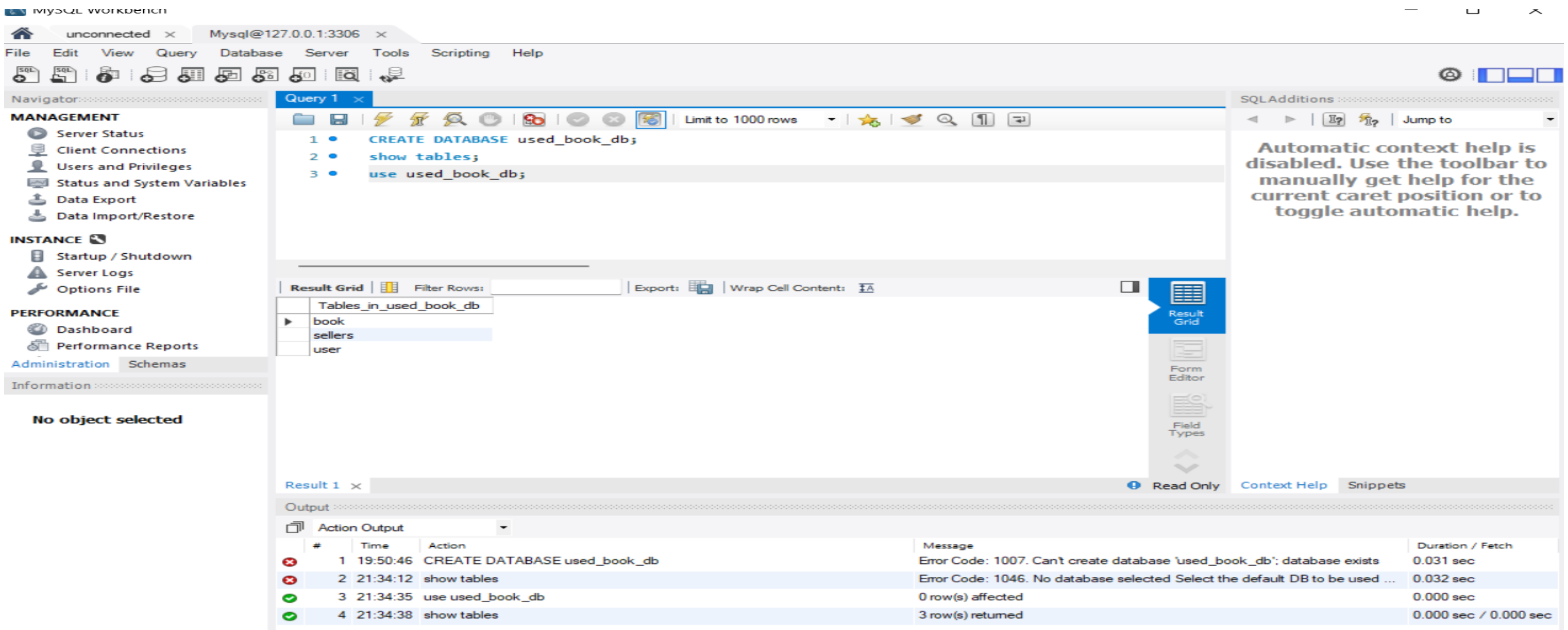
- GET – To retrieve book listings
- POST – To add new books
- PUT – To update existing book details
- DELETE – To remove books from the system

ARCHITECTURE USED

- The application follows Client–Server Architecture.
- The frontend sends requests to the backend server.
- The backend processes the request and interacts with the database.
- Data is exchanged in structured format between layers.
- The MVC pattern ensures separation of concerns.

DATA DICTIONARY

- **TABLES OF DATABASE**



```sql
CREATE DATABASE used_book_db;
show tables;
use used_book_db;
```

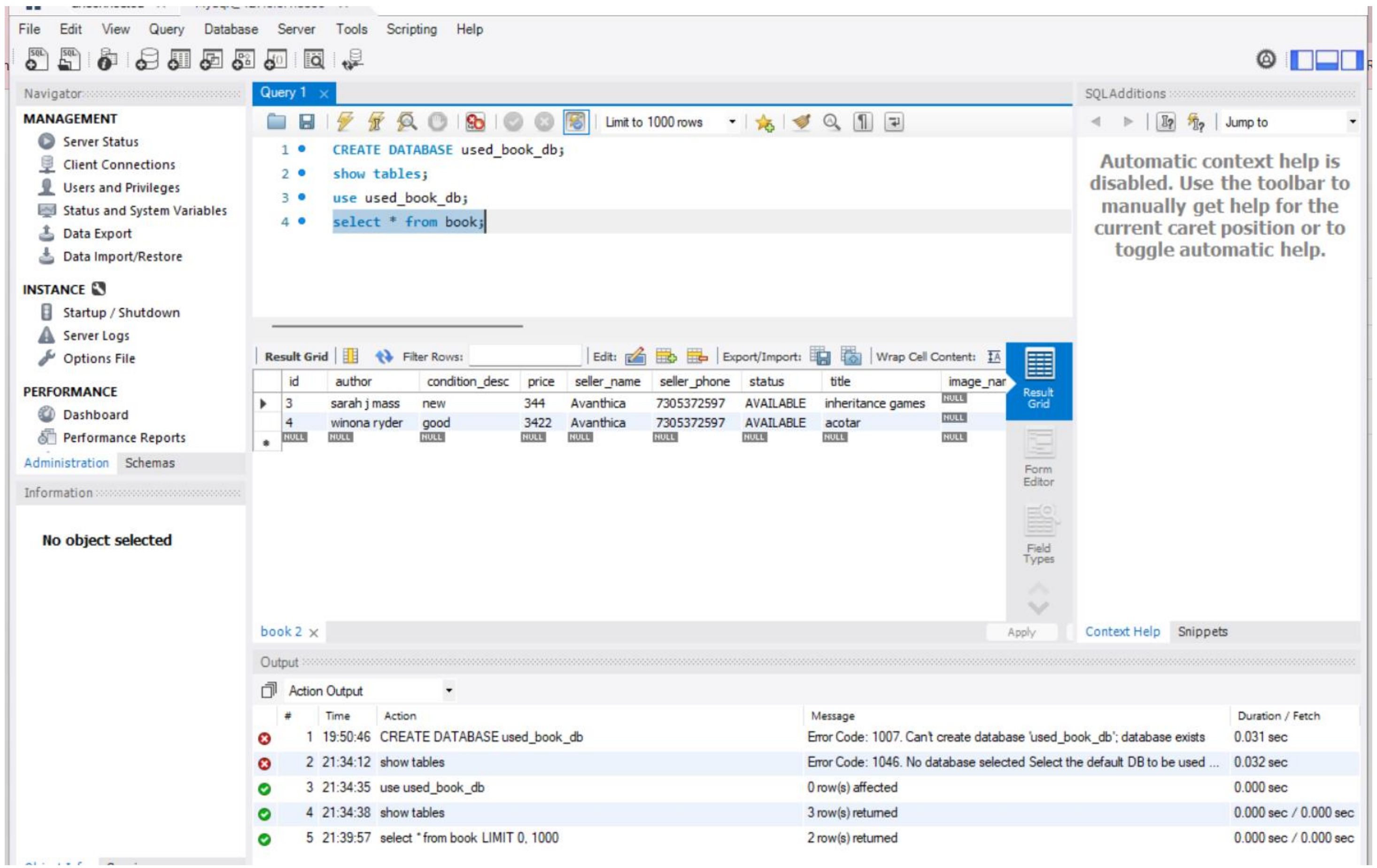
Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

Tables_in_used_book_db
book
sellers
user

No object selected

#	Time	Action	Message	Duration / Fetch
1	19:50:46	CREATE DATABASE used_book_db	Error Code: 1007. Can't create database 'used_book_db'; database exists	0.031 sec
2	21:34:12	show tables	Error Code: 1046. No database selected Select the default DB to be used ...	0.032 sec
3	21:34:35	use used_book_db	0 row(s) affected	0.000 sec
4	21:34:38	show tables	3 row(s) returned	0.000 sec / 0.000 sec

Book database

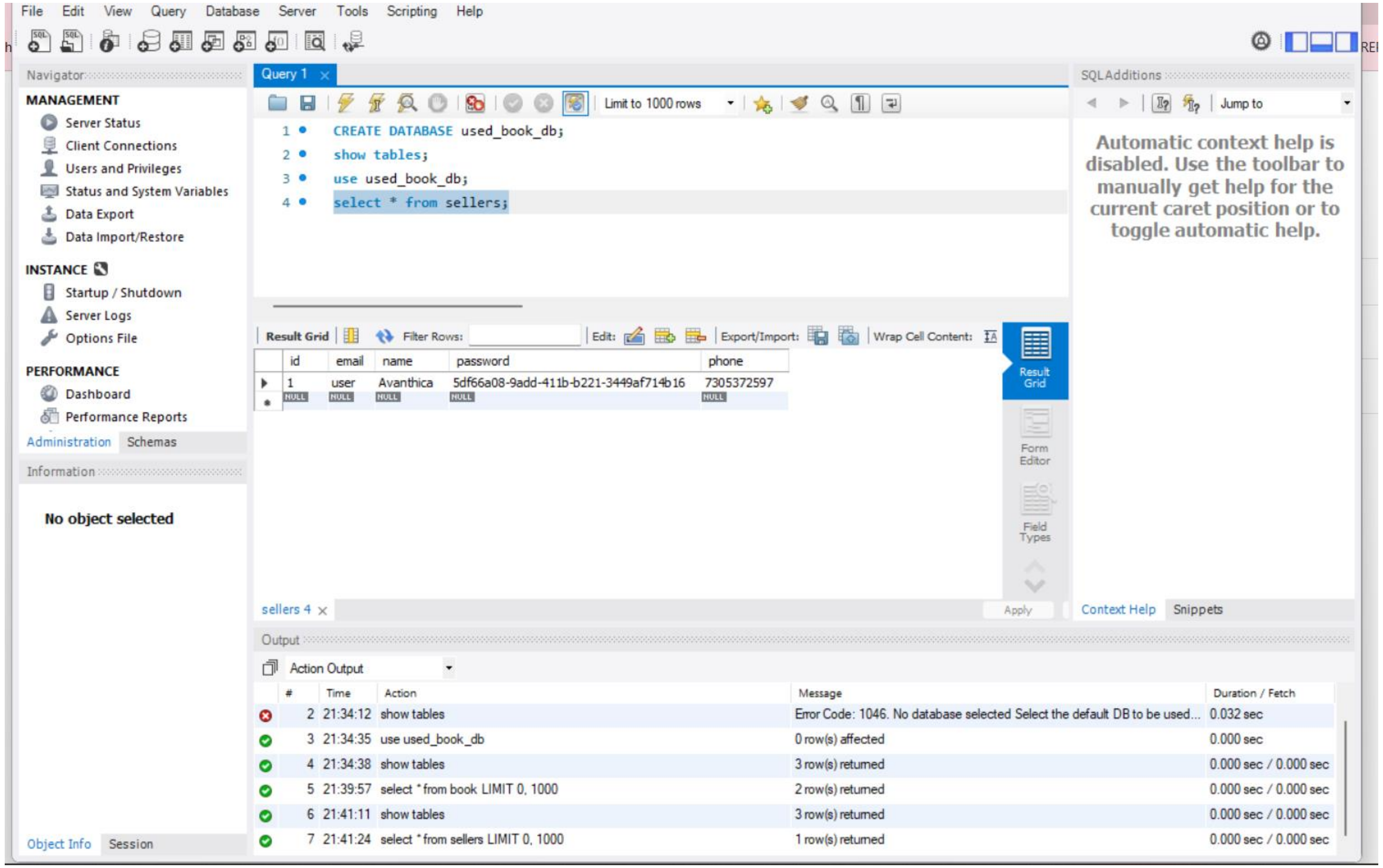


```
CREATE DATABASE used_book_db;
show tables;
use used_book_db;
select * from book;
```

	id	author	condition_desc	price	seller_name	seller_phone	status	title	image_nar
▶	3	sarah j mass	new	344	Avanthica	7305372597	AVAILABLE	inheritance games	NULL
	4	winona ryder	good	3422	Avanthica	7305372597	AVAILABLE	acotar	NULL
*	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

```
   #  Time      Action                            Message                                                               Duration / Fetch
✖  1  19:50:46  CREATE DATABASE used_book_db      Error Code: 1007. Can't create database 'used_book_db'; database exists  0.031 sec
✖  2  21:34:12  show tables                       Error Code: 1046. No database selected Select the default DB to be used ...  0.032 sec
✓  3  21:34:35  use used_book_db                  0 row(s) affected                                                     0.000 sec
✓  4  21:34:38  show tables                       3 row(s) returned                                                     0.000 sec / 0.000 sec
✓  5  21:39:57  select * from book LIMIT 0, 1000  2 row(s) returned                                                     0.000 sec / 0.000 sec
```

Seller database



No object selected

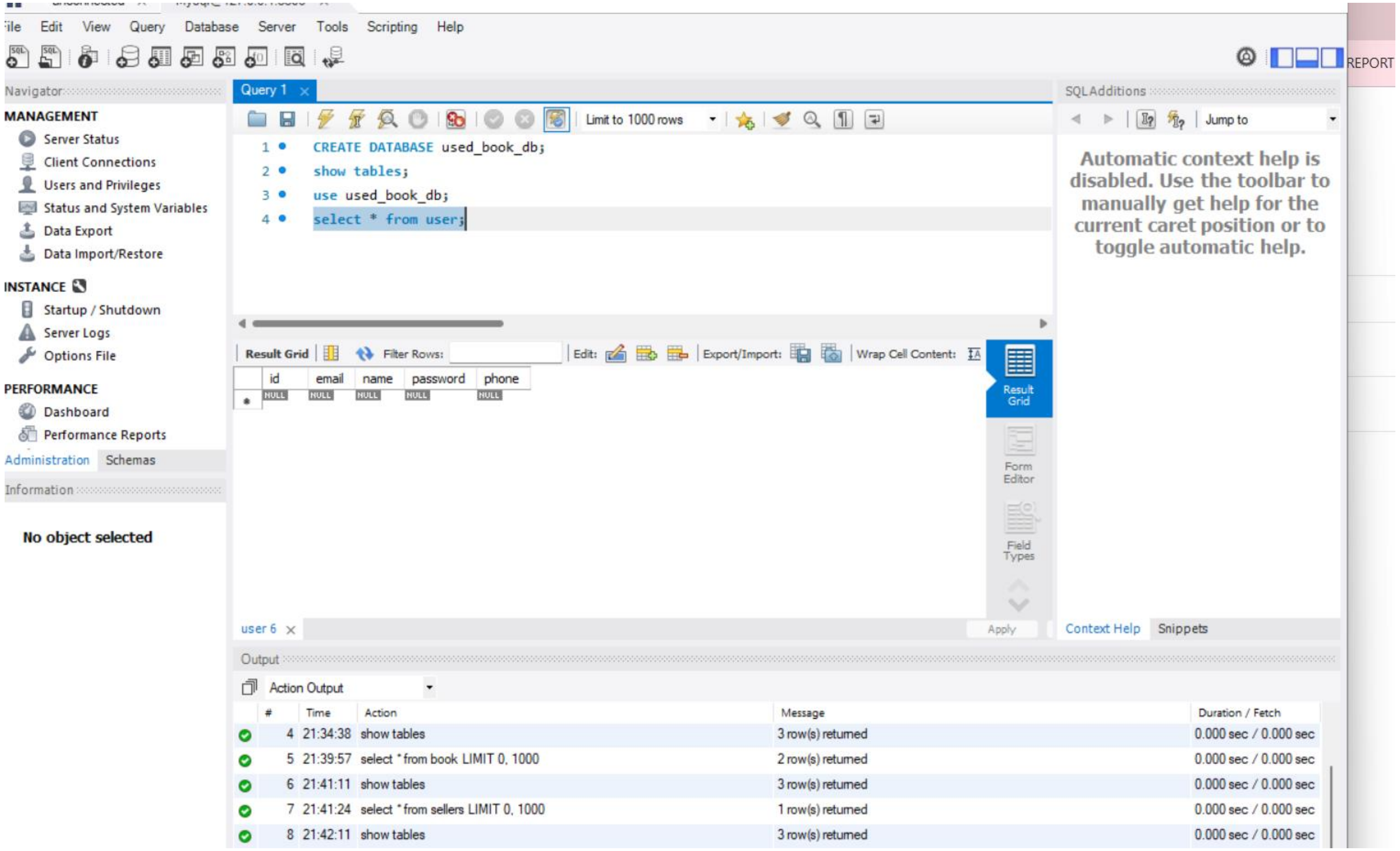
```
CREATE DATABASE used_book_db;
show tables;
use used_book_db;
select * from sellers;
```

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

	id	email	name	password	phone
▶	1	user	Avanthica	5df66a08-9add-411b-b221-3449af714b16	7305372597
●	NULL	NULL	NULL	NULL	NULL

	#	Time	Action	Message	Duration / Fetch
⊗	2	21:34:12	show tables	Error Code: 1046. No database selected Select the default DB to be used...	0.032 sec
✓	3	21:34:35	use used_book_db	0 row(s) affected	0.000 sec
✓	4	21:34:38	show tables	3 row(s) returned	0.000 sec / 0.000 sec
✓	5	21:39:57	select * from book LIMIT 0, 1000	2 row(s) returned	0.000 sec / 0.000 sec
✓	6	21:41:11	show tables	3 row(s) returned	0.000 sec / 0.000 sec
✓	7	21:41:24	select * from sellers LIMIT 0, 1000	1 row(s) returned	0.000 sec / 0.000 sec

User database



```
CREATE DATABASE used_book_db;
show tables;
use used_book_db;
select * from user;
```

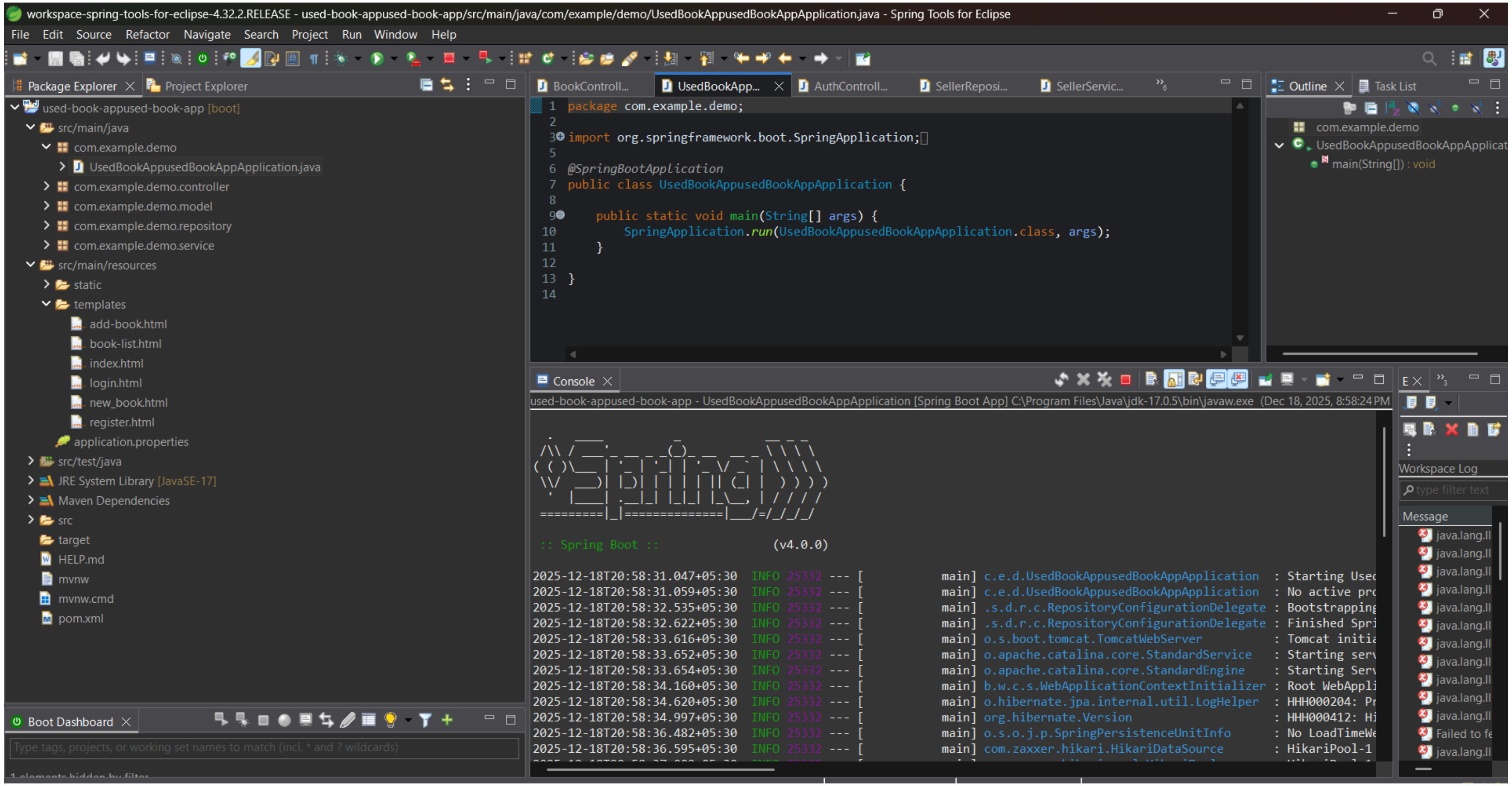
id	email	name	password	phone
NULL	NULL	NULL	NULL	NULL

Automatic context help is disabled. Use the toolbar to manually get help for the current caret position or to toggle automatic help.

No object selected

```
#   Time      Action                                Message            Duration / Fetch
4   21:34:38  show tables                           3 row(s) returned  0.000 sec / 0.000 sec
5   21:39:57  select * from book LIMIT 0, 1000      2 row(s) returned  0.000 sec / 0.000 sec
6   21:41:11  show tables                           3 row(s) returned  0.000 sec / 0.000 sec
7   21:41:24  select * from sellers LIMIT 0, 1000   1 row(s) returned  0.000 sec / 0.000 sec
8   21:42:11  show tables                           3 row(s) returned  0.000 sec / 0.000 sec
```

Main program



workspace-spring-tools-for-eclipse-4.32.2.RELEASE - used-book-appused-book-app/src/main/java/com/example/demo/UsedBookAppusedBookAppApplication.java - Spring Tools for Eclipse

```
used-book-appused-book-app [boot]
  src/main/java
    com.example.demo
      UsedBookAppusedBookAppApplication.java
    com.example.demo.controller
    com.example.demo.model
    com.example.demo.repository
    com.example.demo.service
  src/main/resources
    static
    templates
      add-book.html
      book-list.html
      index.html
      login.html
      new_book.html
      register.html
    application.properties
  src/test/java
  JRE System Library [JavaSE-17]
  Maven Dependencies
  src
  target
  HELP.md
  mvnw
  mvnw.cmd
  pom.xml
```

```java
package com.example.demo;

import org.springframework.boot.SpringApplication;

@SpringBootApplication
public class UsedBookAppusedBookAppApplication {

    public static void main(String[] args) {
        SpringApplication.run(UsedBookAppusedBookAppApplication.class, args);
    }

}
```

Outline: com.example.demo; UsedBookAppusedBookAppApplicat; main(String[]) : void

used-book-appused-book-app - UsedBookAppusedBookAppApplication [Spring Boot App] C:\Program Files\Java\jdk-17.0.5\bin\javaw.exe (Dec 18, 2025, 8:58:24 PM

```
  .   ____          _            __ _ _
 /\\ / ___'_ __ _ _(_)_ __  __ _ \ \ \ \
( ( )\___ | '_ | '_| | '_ \/ _` | \ \ \ \
 \\/  ___)| |_)| | | | | || (_| |  ) ) ) )
  '  |____| .__|_| |_|_| |_\__, | / / / /
 =========|_|==============|___/=/_/_/_/

 :: Spring Boot ::                (v4.0.0)

2025-12-18T20:58:31.047+05:30  INFO 25332 --- [           main] c.e.d.UsedBookAppusedBookAppApplication  : Starting Used
2025-12-18T20:58:31.059+05:30  INFO 25332 --- [           main] c.e.d.UsedBookAppusedBookAppApplication  : No active pro
2025-12-18T20:58:32.535+05:30  INFO 25332 --- [           main] .s.d.r.c.RepositoryConfigurationDelegate : Bootstrapping
2025-12-18T20:58:32.622+05:30  INFO 25332 --- [           main] .s.d.r.c.RepositoryConfigurationDelegate : Finished Spri
2025-12-18T20:58:33.616+05:30  INFO 25332 --- [           main] o.s.boot.tomcat.TomcatWebServer          : Tomcat initia
2025-12-18T20:58:33.652+05:30  INFO 25332 --- [           main] o.apache.catalina.core.StandardService   : Starting serv
2025-12-18T20:58:33.654+05:30  INFO 25332 --- [           main] o.apache.catalina.core.StandardEngine    : Starting Serv
2025-12-18T20:58:34.160+05:30  INFO 25332 --- [           main] b.w.c.s.WebApplicationContextInitializer : Root WebAppli
2025-12-18T20:58:34.620+05:30  INFO 25332 --- [           main] o.hibernate.jpa.internal.util.LogHelper  : HHH000204: Pr
2025-12-18T20:58:34.997+05:30  INFO 25332 --- [           main] org.hibernate.Version                    : HHH000412: Hi
2025-12-18T20:58:36.482+05:30  INFO 25332 --- [           main] o.s.o.j.p.SpringPersistenceUnitInfo      : No LoadTimeWe
2025-12-18T20:58:36.595+05:30  INFO 25332 --- [           main] com.zaxxer.hikari.HikariDataSource       : HikariPool-1
```

```
java.lang.Il
java.lang.Il
java.lang.Il
java.lang.Il
java.lang.Il
java.lang.Il
java.lang.Il
java.lang.Il
java.lang.Il
java.lang.Il
java.lang.Il
Failed to fe
java.lang.Il
```

Video link

- https://drive.google.com/file/d/1k47wtXYKe4fqcQSa19U9toxM-sTGrTRS/view?usp=sharing