Relevel
by Unacademy

Installation Instructions

In order to build the feature please follow the below-mentioned instructions:

Ensure that java, postman and intellij/eclipse are already installed on your system before starting the development. Install spring-boot-suite from https://spring.io/projects/spring-boot.

The tech stack required is Java and Spring Boot. The preferred IDE is Intellij/Eclipse to import the project directly. Postman tool to test the REST APIs.
Ensure that maven and .m2 etc are configured on local. While accessing the code, ignore the _MACOSX folder. The code is present in the 'bookMyShowClone-main' folder.

To check the hosted application in the browser, they can search for http://localhost:8000/ in their browser or postman app after running the main class in IDE.

For the persistence of data, H2 database should be used which is an in-memory database and dependency for the same is already added in the POM. If you are comfortable with any other in-memory database like SQLite, feel free to add the dependency and use that.

H2 database

The database is automatically configured with a sample table BILLIONAIRES. H2 console can be accessed with the following credentials.

```
username=sa
password=password
```

Console link - http://localhost:8080/h2-console
Default JDBC URL: jdbc:h2:mem:testdb
How to connect to JDBC:
http://www.h2database.com/html/tutorial.html#connecting_using_jdbc

Submission Instructions

Code Submission:

1. Compress the code on the local system in the form of a *.zip file.
2. Upload the code on your personal google drive in a folder titled - "Name_BD_<Round Name> App"

3. Don’t forget to change the permissions of the folder to ‘Anyone with the link can edit’.

Loom video submission:

1. Create an account on Loom.
2. Go through the quick tutorial on how to record loom videos.
3. Create a Loom video (while screensharing) covering the following points:
    a. Show the functionality of the app you have created i.e demo of the working APIs through a command line. (1 min)
    b. Run through the key parts of your code explaining the core logic and how you organized the code. (2 min)
    c. Explain your problem-solving approach (what logic you have used and why). (2 min)
4. Please keep your explanation to under 5 mins only.
5. Avoid too much jargon and explain your app in a simple and clear manner.