

LIBRARY PROJECT DOCUMENTATION

Tools Used

IntelliJ IDEA 2022.1.3
MySQL Server 8
Java 18

Downloading and Configuring The Source Code

The project code can be downloaded by using the following command;
`git clone https://github.com/isikozsoy/library`

The configuration will be done automatically.

Setting Up The Database

Make sure that the MySQL Server is running on your machine, it can be downloaded from the link below;

<https://dev.mysql.com/downloads/windows/installer/8.0.html>

Create MySQL Connection and write the necessary information (`hostname`, `port`, `username` and `password`) into `application.properties` file, which is under the resource folder.

Then, create a new schema called `libraryapp` into the MySQL connection that is created. Provide the name of the schema into `application.properties` file as shown below.

```
spring.datasource.driver-class-name=com.mysql.cj.jdbc.Driver
spring.datasource.url=jdbc:mysql://127.0.0.1:3306/libraryapp
spring.datasource.username=root
spring.datasource.password=root
spring.jpa.show-sql=true
spring.jpa.hibernate.ddl-auto=update
spring.jpa.properties.hibernate.dialect=org.hibernate.dialect.MySQL5Dialect
```

Development Set Up

Using Swagger for Testing

Swagger framework allows developers to create interactive, machine and human-readable API documentation. The documentation can be accessed by the following url after building and running the project;

<http://localhost:8080/swagger-ui.html>

localhost:8080/swagger-ui.html

swagger default (/v2/api-docs) api_key Explore

Online Library

"Online Library build by IO"

author-controller : Author Controller

Show/Hide | List Operations | Expand Operations

POST	/authorAPI/addAuthor	addAuthor
GET	/authorAPI/authorById/{id}	findAuthorById
GET	/authorAPI/authorByName/{fullName}	findAuthorByFullName
GET	/authorAPI/authorByNationality/{nationality}	findAuthorByNationality
GET	/authorAPI/authors	findAllAuthors
DELETE	/authorAPI/deleteAuthor/{id}	deleteBook
PUT	/authorAPI/updateAuthor	updateAuthor

book-controller : Book Controller

Show/Hide | List Operations | Expand Operations

[BASE URL: / , API VERSION: 1.0.0]

author-controller : Author Controller

Show/Hide | List Operations | Expand Operations

POST	/authorAPI/addAuthor	addAuthor
GET	/authorAPI/authorById/{id}	findAuthorById

Response Class (Status 200)

Model | Model Schema

```
{
  "fullName": "string",
  "id": 0,
  "nationality": "string"
}
```

Response Content Type

Parameters

Parameter	Value	Description	Parameter Type	Data Type
id	(required)	Id	path	integer

Response Messages

HTTP Status Code	Reason	Response Model	Headers
401	Unauthorized		
403	Forbidden		
404	Not Found		

Try it out!

As can be seen in the screenshots above, the user can select the controller and a request, the details are shown such as parameters, response type.