

Ejercicio Práctico

IMPLEMENTACIÓN DE UNA API HTTP CON SOCKETS Y MYSQL
EN JAVA

LingCheng Jiang
2ºDAM T | PGV

Contenido

1. Estructura del Proyecto	2
2. Pruebas del proyecto.....	2
2.1. GET.....	2
2.2. POST.....	3
2.3. PUT.....	4
2.4. DELETE	5

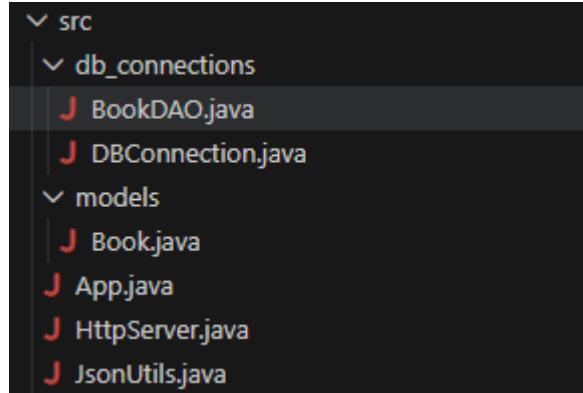
1. Estructura del Proyecto

Para que el proyecto funcione, debe estar instalado MySQL y tener la base de datos y usuario creado.

Simplemente con ejecutar una vez “WebSocket.sql” se genera la BD y usuario con datos de prueba.

El proyecto en sí, está compuesto de 6 archivos .java, donde el mayor peso recae en HttpServer.java y BookDAO.java

Siendo HttpServer.java el encargado de enrutar, y contactar con las peticiones y mandar a realizarlas a la base de datos a BookDAO.java.



2. Pruebas del proyecto

2.1. GET

```
[{"id": 1, "title": "To Kill a Mockingbird", "author": "Harper Lee", "year": 1960}, {"id": 2, "title": "1984", "author": "George Orwell", "year": 1949}, {"id": 3, "title": "The Great Gatsby", "author": "F. Scott Fitzgerald", "year": 1925}, {"id": 4, "title": "The Catcher in the Rye", "author": "J.D. Salinger", "year": 1951}, {"id": 5, "title": "Moby DICK", "author": "Herman Melville", "year": 1851}]
```

2.2. POST

The screenshot shows the Postman application interface. At the top, there are tabs for 'GET New Request', 'POST New Request' (which is selected), 'PUT New Request', and 'DEL New Request'. Below the tabs, the URL is set to 'TestingWebSockets / New Request' via 'http'. The main area shows a POST request to 'http://localhost:8080/books'. The 'Body' tab is active, showing raw JSON input:

```
1 {  
2   "title": "Test Posting",  
3   "author": "LINGC",  
4   "year": 2025  
5 }
```

The response status is '201 OK' with a response time of 30 ms and a size of 131 B. The response body is also shown in JSON format:

```
1 {  
2   "id": 6,  
3   "title": "Test Posting",  
4   "author": "LINGC",  
5   "year": 2025  
6 }
```

Below this, another section shows a GET request to 'http://localhost:8080/books'. The 'Params' tab is active. The query parameters table has one row:

Key	Value	Description
Key	Value	Description

The response status is '200 OK' with a response time of 13 ms and a size of 496 B. The response body is a list of books:

```
7 },  
8 {  
9   "id": 2,  
10  "title": "1984",  
11  "author": "George Orwell",  
12  "year": 1949  
13 },  
14 {  
15   "id": 3,  
16   "title": "The Great Gatsby",  
17   "author": "F. Scott Fitzgerald",  
18   "year": 1925  
19 },  
20 {  
21   "id": 4,  
22   "title": "The Catcher in the Rye",  
23   "author": "J.D. Salinger",  
24   "year": 1951  
25 },  
26 {  
27   "id": 5,  
28   "title": "Moby DICK",  
29   "author": "Herman Melville",  
30   "year": 1851  
31 },  
32 {  
33   "id": 6,  
34   "title": "Test Posting",  
35   "author": "LINGC",  
36   "year": 2025  
37 }  
38 ]
```

2.3. PUT

The screenshot shows the Postman application interface with two requests:

PUT Request:

- Method: PUT
- URL: <http://localhost:8080/books/6>
- Body (raw JSON):

```
1 {
2   "title": "UpdatedTesting",
3   "author": "lcj",
4   "year": 2025
5 }
```

Response: 200 OK | 20 ms | 72 B

GET Request:

- Method: GET
- URL: <http://localhost:8080/books>
- Params:

Key	Value	Description
Key	Value	Description

Response: 200 OK | 21 ms | 496 B

The body of the GET response shows a list of books:

```
7 },
8 {
9   "id": 2,
10  "title": "1984",
11  "author": "George Orwell",
12  "year": 1949
13 },
14 {
15   "id": 3,
16   "title": "The Great Gatsby",
17   "author": "F. Scott Fitzgerald",
18   "year": 1925
19 },
20 {
21   "id": 4,
22   "title": "The Catcher in the Rye",
23   "author": "J.D. Salinger",
24   "year": 1951
25 },
26 {
27   "id": 5,
28   "title": "Moby DICK",
29   "author": "Herman Melville",
30   "year": 1851
31 },
32 {
33   "id": 6,
34   "title": "UpdatedTesting",
35   "author": "lcj",
36   "year": 2025
37 }
38 ]
```

2.4. DELETE

The screenshot shows the Postman application interface with two requests displayed.

DELETE Request:

- Method: **DELETE**
- URL: **http://localhost:8080/books/6**
- Headers: **Content-Type: application/json**
- Body: **{} JSON**
- Response: Status 200 OK, 17 ms, 72 B. Body: **[{"id": 1, "title": "To Kill a Mockingbird", "author": "Harper Lee", "year": 1960}, {"id": 2, "title": "1984", "author": "George Orwell", "year": 1949}, {"id": 3, "title": "The Great Gatsby", "author": "F. Scott Fitzgerald", "year": 1925}, {"id": 4, "title": "The Catcher in the Rye", "author": "J.D. Salinger", "year": 1951}, {"id": 5, "title": "Moby DICK", "author": "Herman Melville", "year": 1851}]**

GET Request:

- Method: **GET**
- URL: **http://localhost:8080/books**
- Headers: **Content-Type: application/json**
- Body: **{} JSON**
- Response: Status 200 OK, 15 ms, 435 B. Body: **[{"id": 1, "title": "To Kill a Mockingbird", "author": "Harper Lee", "year": 1960}, {"id": 2, "title": "1984", "author": "George Orwell", "year": 1949}, {"id": 3, "title": "The Great Gatsby", "author": "F. Scott Fitzgerald", "year": 1925}, {"id": 4, "title": "The Catcher in the Rye", "author": "J.D. Salinger", "year": 1951}, {"id": 5, "title": "Moby DICK", "author": "Herman Melville", "year": 1851}]**