

# Ejercicio Práctico

---

IMPLEMENTACIÓN DE UNA API HTTP CON SOCKETS Y MYSQL  
EN JAVA

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

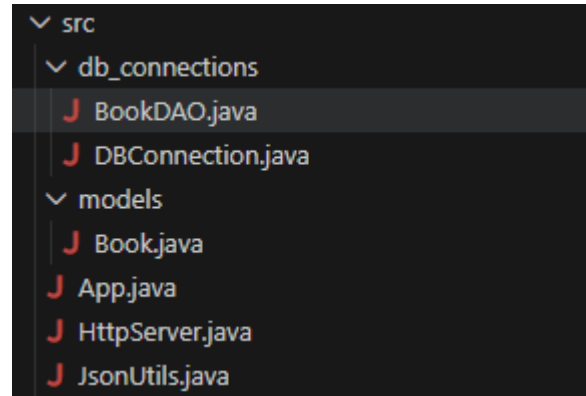
# 1. Estructura del Proyecto

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.

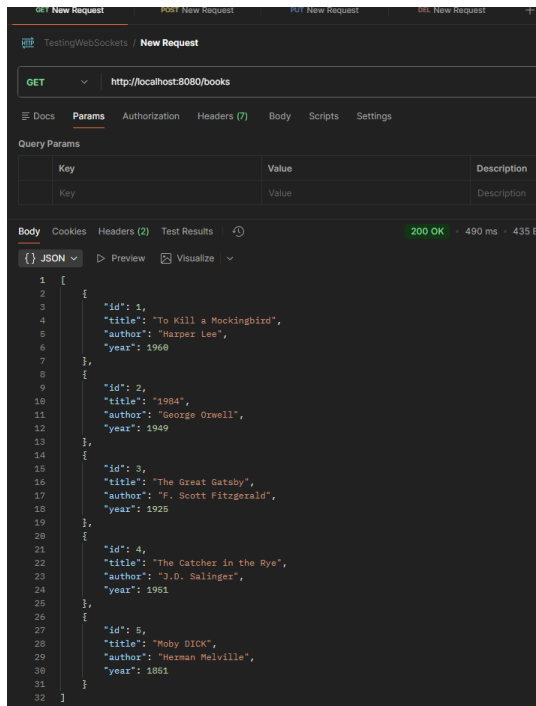
El proyecto en sí, está compuesto de 6 archivos .java, donde el mayor peso recae en HttpServer.java y BookDAO.java

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

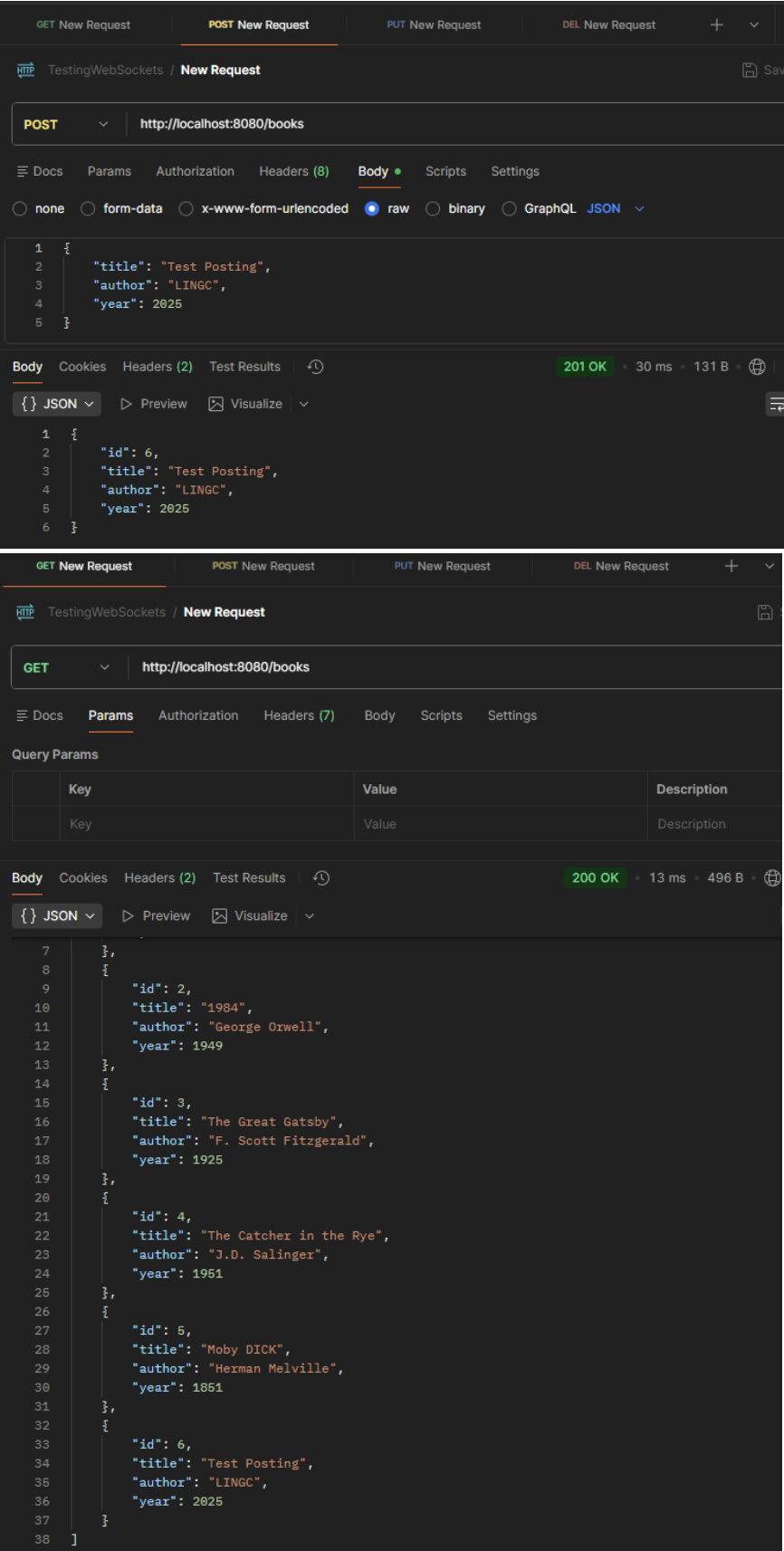


## 2. Pruebas del proyecto

### 2.1. GET



## 2.2. POST



## 2.3. PUT

The screenshot shows a REST client interface with the 'PUT' method selected. The URL is `http://localhost:8080/books/6`. The 'Body' tab is active, showing a JSON object: `{ "title": "UpdatedTesting", "author": "lcj", "year": 2025 }`. The status bar indicates a `200 OK` response with a 20 ms duration and 72 B of data.

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

Body Cookies Headers (2) Test Results 200 OK • 20 ms • 72 B

{ } JSON Preview Visualize

```
1 {}
```

The screenshot shows a REST client interface with the 'GET' method selected. The URL is `http://localhost:8080/books`. The 'Body' tab is active, showing a JSON array of book objects. The status bar indicates a `200 OK` response with a 21 ms duration and 496 B of data.

Query Params

Key	Value	Description
Key	Value	Description

```
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 ]
```

Body Cookies Headers (2) Test Results 200 OK • 21 ms • 496 B

{ } JSON Preview Visualize

## 2.4. DELETE

GET New RequestPOST New RequestPUT New RequestDEL New Request

TestingWebSockets / New Request

DELETEhttp://localhost:8080/books/6

DocsParamsAuthorizationHeaders (6)BodyScriptsSettings

Query Params

Key	Value	Description
Key	Value	Description

BodyCookiesHeaders (2)Test Results200 OK17 ms72 B

JSONPreviewVisualize

1 {}

GET New RequestPOST New RequestPUT New RequestDEL New Request

TestingWebSockets / New Request

GEThttp://localhost:8080/books

DocsParamsAuthorizationHeaders (7)BodyScriptsSettings

Query Params

Key	Value	Description
Key	Value	Description

BodyCookiesHeaders (2)Test Results200 OK15 ms435 B

JSONPreviewVisualize

1 [2 { "id": 1, "title": "To Kill a Mockingbird", "author": "Harper Lee", "year": 1960 }, 3 { "id": 2, "title": "1984", "author": "George Orwell", "year": 1949 }, 4 { "id": 3, "title": "The Great Gatsby", "author": "F. Scott Fitzgerald", "year": 1925 }, 5 { "id": 4, "title": "The Catcher in the Rye", "author": "J.D. Salinger", "year": 1951 }, 6 { "id": 5, "title": "Moby DICK", "author": "Herman Melville", "year": 1851 } ]