|  |  |
| --- | --- |
| **Topic** | SWAGGER |
| **Document Name** | SWAGGER |
|  |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **Document Difficulty Level** | | | |
| **Beginner** | **Junior** | **Senior** | **Expert** |
| □ | ■ | □ | □ |

# Document History

|  |  |  |  |
| --- | --- | --- | --- |
| Date | Author | Ver | Comments |
| 05.03.2025 | Murat Kağan Temel | 1.0 | Initial Draft |

# SWAGGER

## Exercise SWAGGER:

**Definiton:** Create a layered Spring Boot RESTful application that provides basic CRUD operations for managing Customer data. The application should include OpenAPI-compliant API documentation using the springdoc-openapi-starter-webmvc-ui dependency. This dependency also enables Swagger UI, so no additional configuration is required.

You may choose to simulate database operations using in-memory data structures such as Map or List. Using a real database is not mandatory.

The application should expose endpoints for the following operations:

* **Create a new customer**
* **Retrieve a customer by ID**
* **Retrieve all customers**
* **Update an existing customer**
* **Delete a customer**

After completing the implementation, open the Swagger UI in your browser and share screenshots of the generated API documentation and the available endpoints. A sample CustomerDTO (DTO refers to Data Transfer Object) is shared in the next page.

public class CustomerDTO {

@Schema(description = "Unique ID of the customer", example = "1")

private Long id;

@NotBlank

@Schema(description = "Full name of the customer", example = "John Doe")

private String name;

@Email

@Schema(description = "Email address", example = "john.doe@example.com")

private String email;

}

## SWAGGER:

**Your Answer:**

**EGE DiREK**

**Hello There! 👋**

For today task: Create a layered Spring Boot RESTful application that provides basic CRUD operations for managing Customer data. The application should include OpenAPI-compliant API documentation using the springdoc-openapi-starter-webmvc-ui dependency. This dependency also enables Swagger UI, so no additional configuration is required.

You may choose to simulate database operations using in-memory data structures such as Map or List. Using a real database is not mandatory.

The application should expose endpoints for the following operations: • Create a new customer

• Retrieve a customer by ID

• Retrieve all customers

• Update an existing customer

• Delete a customer

After completing the implementation, open the Swagger UI in your browser and share screenshots of the generated API documentation and the available endpoints. A sample CustomerDTO (DTO refers to Data Transfer Object) is shared in the next page.

**Steps:**

**1- Create DTO, serviceses, repositiories and controller (REST API) file for customer.**

**2- Create a main file for run the app (CustomerManagementApiApplication).**

**3- Add these dependencies for Swagger UI and lombok for auto generate repetitive codes.**

<dependency>

<groupId>org.projectlombok</groupId>

<artifactId>lombok</artifactId>

<version>1.18.38</version>

<optional>true</optional>

<scope>provided</scope>

</dependency>

<dependency>

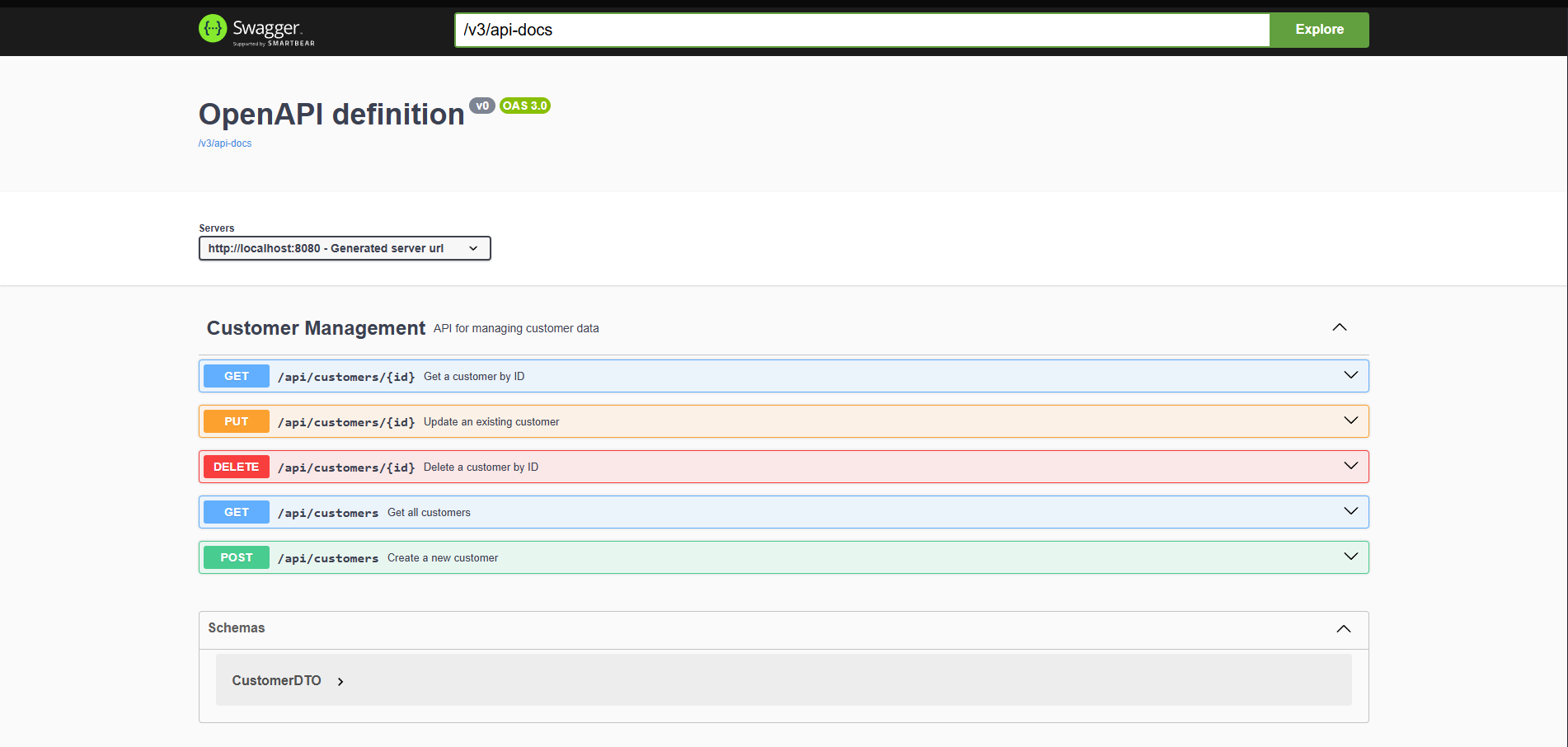
<groupId>org.springdoc</groupId>

<artifactId>springdoc-openapi-starter-webmvc-ui</artifactId>

<version>2.5.0</version>

</dependency>

**4- Change your port to your local host and run to access Swagger UI**

[](https://private-user-images.githubusercontent.com/209494583/462616687-237e9fed-1786-4267-a96a-94ba3486a600.png?jwt=eyJhbGciOiJIUzI1NiIsInR5cCI6IkpXVCJ9.eyJpc3MiOiJnaXRodWIuY29tIiwiYXVkIjoicmF3LmdpdGh1YnVzZXJjb250ZW50LmNvbSIsImtleSI6ImtleTUiLCJleHAiOjE3NTE2NTYyMDYsIm5iZiI6MTc1MTY1NTkwNiwicGF0aCI6Ii8yMDk0OTQ1ODMvNDYyNjE2Njg3LTIzN2U5ZmVkLTE3ODYtNDI2Ny1hOTZhLTk0YmEzNDg2YTYwMC5wbmc_WC1BbXotQWxnb3JpdGhtPUFXUzQtSE1BQy1TSEEyNTYmWC1BbXotQ3JlZGVudGlhbD1BS0lBVkNPRFlMU0E1M1BRSzRaQSUyRjIwMjUwNzA0JTJGdXMtZWFzdC0xJTJGczMlMkZhd3M0X3JlcXVlc3QmWC1BbXotRGF0ZT0yMDI1MDcwNFQxOTA1MDZaJlgtQW16LUV4cGlyZXM9MzAwJlgtQW16LVNpZ25hdHVyZT00NjQ5N2FkNjI4M2RjZDhiNWNiNDY3MTM4MmIyN2NhMTUyNjhiMGNlZDBjYzEyMTY0YzIyMDk5ZGYxZDE5MjViJlgtQW16LVNpZ25lZEhlYWRlcnM9aG9zdCJ9.FU19ayKg3Yr5xNl4vdu4l8WBCVWfZknlOz-UF_dJ_Jc)