Project Setup and Execution Guide

1. Build the JAR File

1. Navigate to the Root Directory of the Project

Open a terminal or command prompt and navigate to the root folder of spring boot project lms where the `pom.xml` file is located.

2. Run the Maven Command to Build the JAR

In the root directory of the project, run the following command to clean, install, and skip tests during the build process:

mvn clean install -DskipTests=true

- This will create a `.jar` file with the name of the project.

---

2. Build the Docker Image

1. Navigate to the Folder Containing the Dockerfile

Make sure you're in the root directory of lms project where the `Dockerfile` is located.

2. Build the Docker Image

Run the following Docker command to build the image

docker build -t lms-image:latest .

---

3. Run PostgreSQL & Spring Boot Containers

1. Navigate to the Folder Containing `env.yml`

Open a terminal where the `env.yml` (Docker Compose configuration file) is located.

2. Run Docker Compose to Start Containers

Execute the following command to start PostgreSQL and Spring Boot containers in detached mode:

docker-compose -f env.yml up -d

This will start the containers based on the configuration in `env.yml` and run them in the background.

---

4. Send the Register and Login Requests

1. Send a Register Request

Use Postman or another API client to send a POST request to the registration endpoint:

- POST URL: `http://localhost:8080/users/register`

- Body (raw JSON):

```
json
{
  "email": "marwa4@gmail.com",
  "password": "12345678",
  "name": "marwa"
}
```

2. Send a Login Request

After registration, send a POST request to the login endpoint to get a JWT token:

- POST URL: `http://localhost:8080/login`

- Body (raw JSON):

```
json
{
  "email": "marwa4@gmail.com",
  "password": "12345678"
}
```

3. Retrieve the Token

The login response will contain a JWT token. Copy this token as you will need it in the next steps.

---

5. Install Python Libraries

1. Install the Required Libraries

Run the following command to install the necessary Python libraries:

pip install requests faker pandas

---

6. Run the Python Script

1. Modify the Python Script

Insert the token you received from the login response into the Python script. Replace `your_jwt_token` with the actual token:

```
headers = {
    "Authorization": "Bearer your_jwt_token",
    "Content-Type": "application/json"
}
```

2. Run the Python Script

Once you've modified the script, run it to interact with the API and perform the necessary tasks.

This is the step-by-step guide to setting up and executing the project.