

README - Traini8 Backend Project Setup Instructions

1. Project Overview

This is a Spring Boot project for Traini8, a registry for Government-funded Training Centers.

It contains REST APIs to create and list training centers with proper validations and database storage.

2. Prerequisites

- Java 17 or above
- Maven
- MySQL Database
- Postman (for API testing)
- IDE (IntelliJ IDEA / Eclipse / VS Code)

3. Project Setup Instructions

Step 1: Clone the Repository or Unzip the project folder.

Step 2: Open the project in your preferred IDE.

Step 3: Create a MySQL database named 'traini8db'.

Step 4: Update application.properties file with your MySQL credentials.

Example: `spring.datasource.url=jdbc:mysql://localhost:3306/traini8db`
`spring.datasource.username=root` `spring.datasource.password=your_password`

Step 5: Run the application using your IDE or via terminal using: `mvn spring-boot:run`

Step 6: The application will start at: `http://localhost:8082`

4. API Endpoints

POST / api/training-centers/createtrainingcenter

- Description: Create and save a new training center.
- Request Body: JSON format

{

"centerName": "ABC Skills Center",
"centerCode": "ABC123XY7890",
"address": {

```
"detailedAddress": "123 Street, Near Metro",  
"city": "Bangalore",  
"state": "Karnataka",  
"pincode": "560001"  
},
```

```
"studentCapacity": 200,  
"coursesOffered": ["Java", "Python"],  
"contactEmail": "abc@example.com",  
"contactPhone": "9876543210"  
}
```

GET /api/training-centers/ alltrainingcenter

- Description: Retrieve all stored training centers.
- Response: List of training centers in JSON format.

5. Notes

- Validation is handled through annotations.
- Exception handling is done globally using @ControllerAdvice.
- 'createdOn' timestamp is auto-generated by the server.
- Ensure the database is running before starting the application.

6. Optional Enhancements

- You can add filtering options by city and state in the GET API.
- You may implement pagination for large data.

7. Contact

For any queries, contact your project coordinator.

8. Swagger API Documentation

- Swagger UI is integrated for easy testing and documentation of all APIs.
- Once the application is running, open your browser and go to:

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

- You can explore all available endpoints, request/response formats, and test the APIs directly from the browser.

Example:

POST /api/centers → Create Training Center

GET /api/centers → List All Training Centers