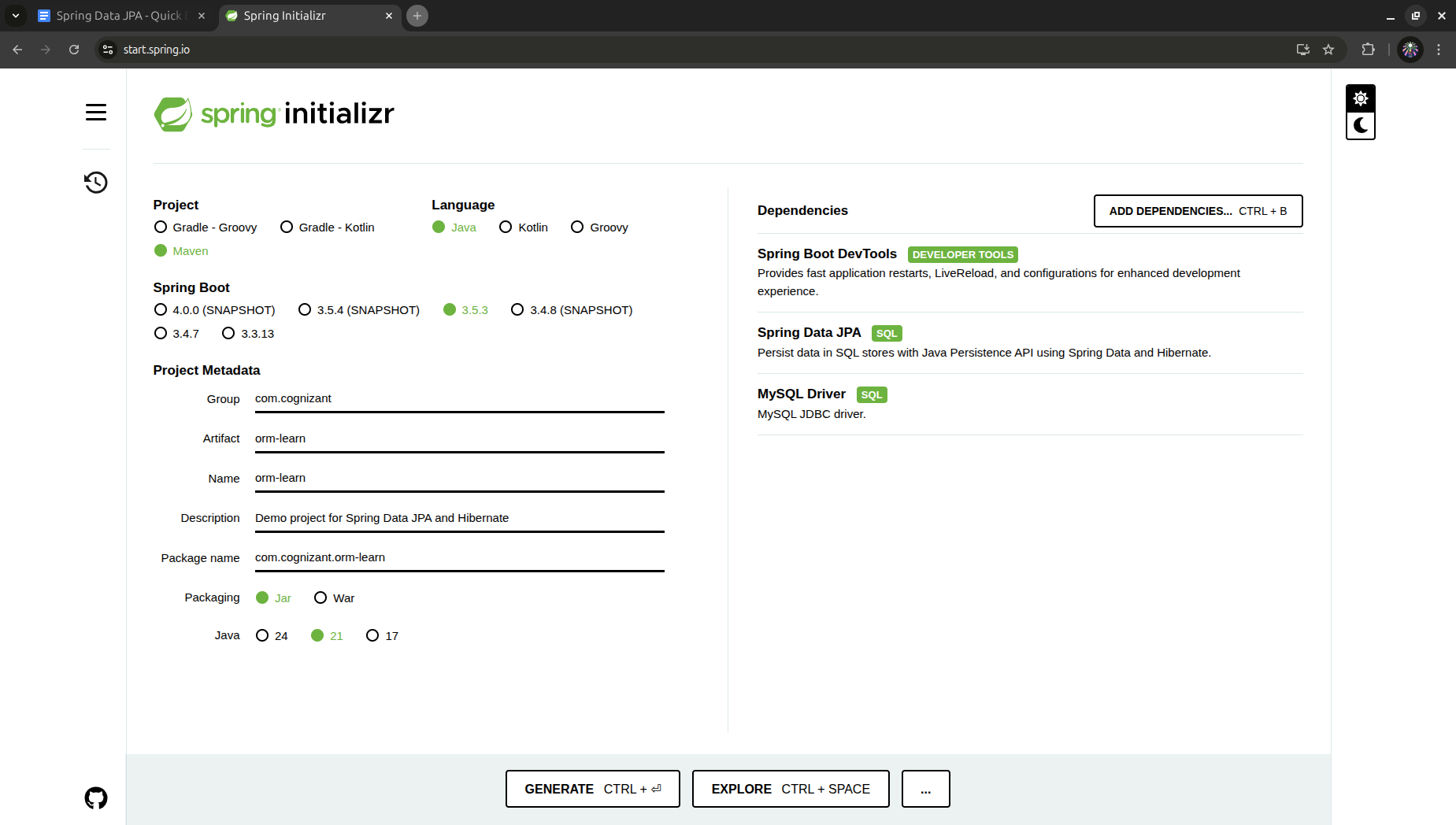
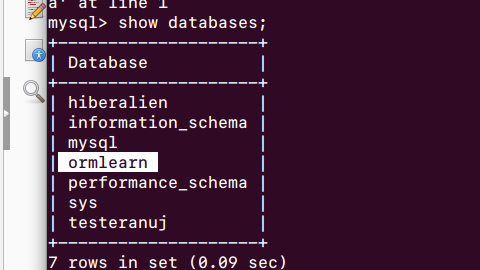
| *Spring Data JPA - Quick Example* |
| --- |

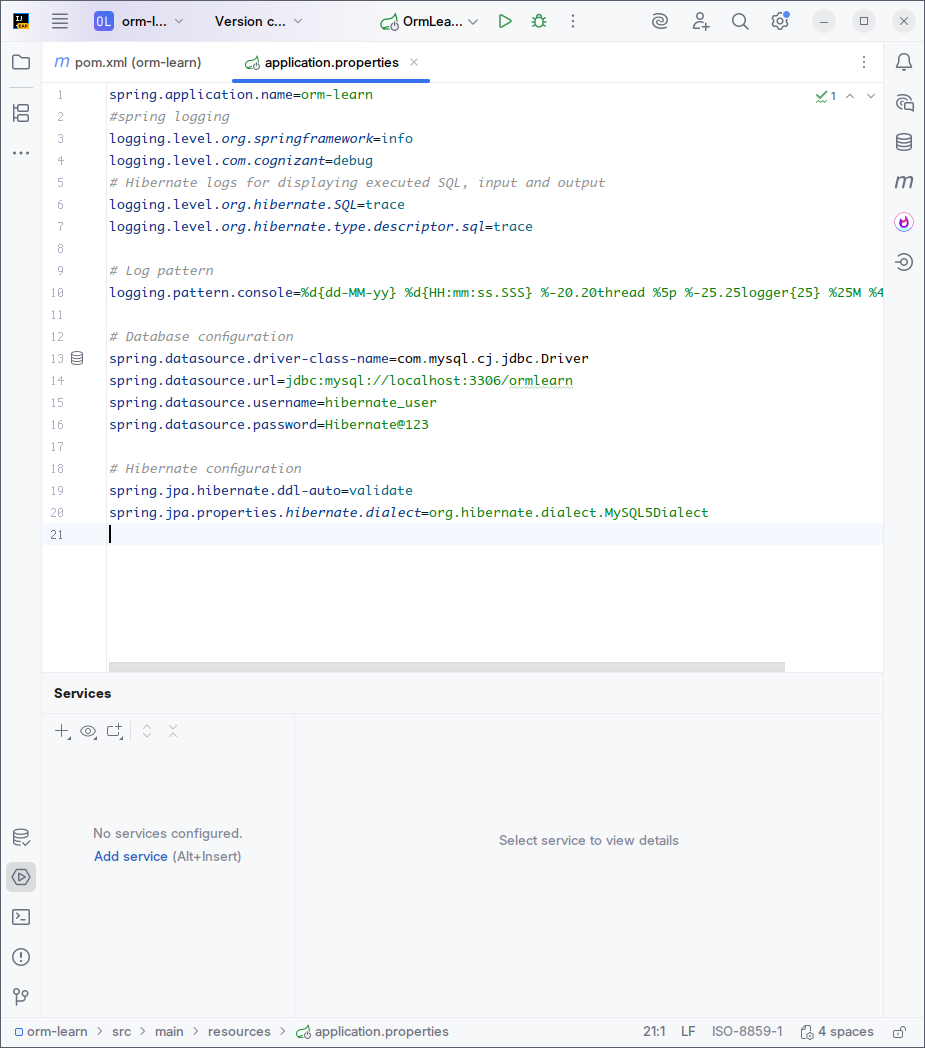


Created schema by using sql query : “create schema ormlearn;”

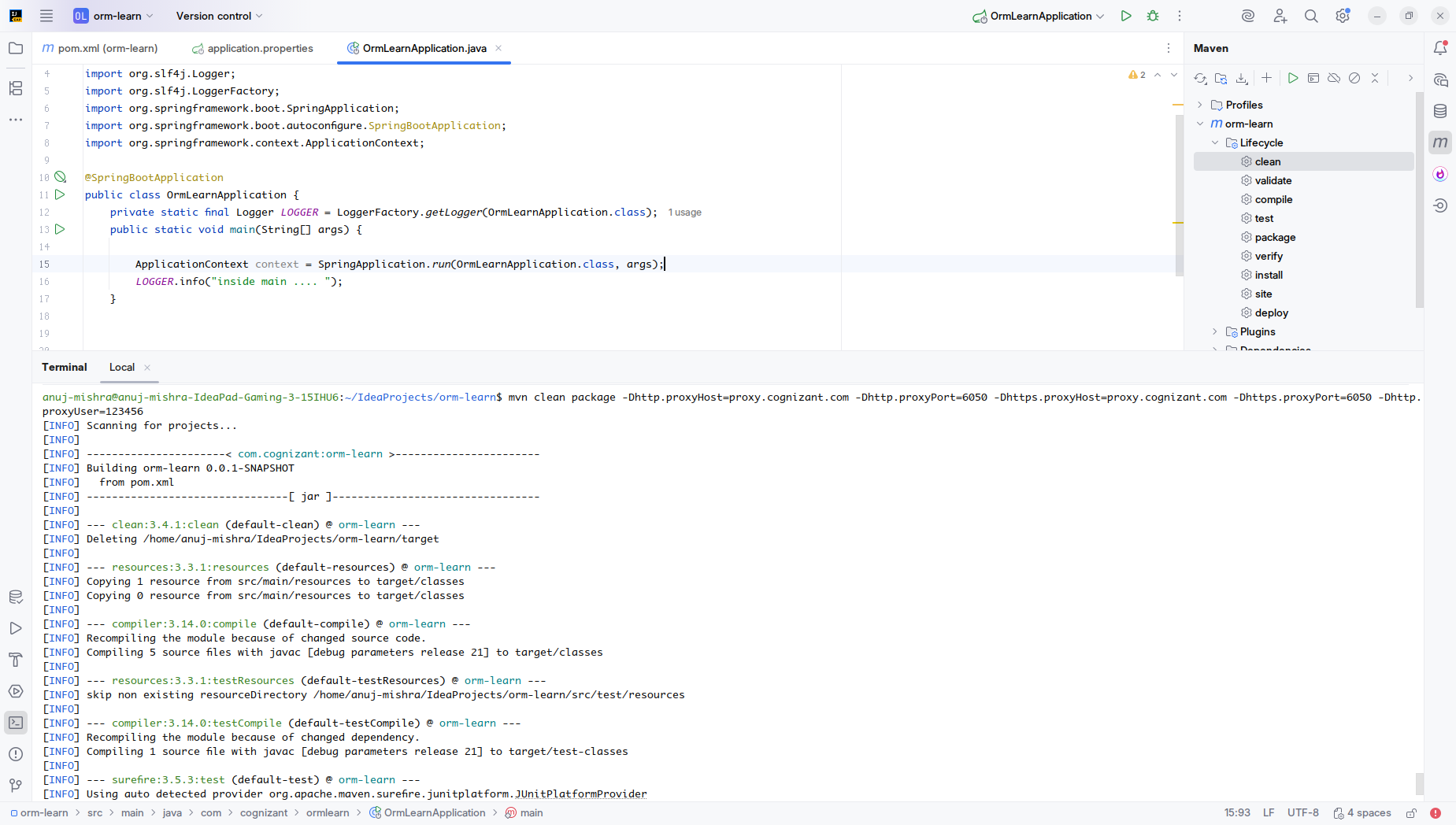


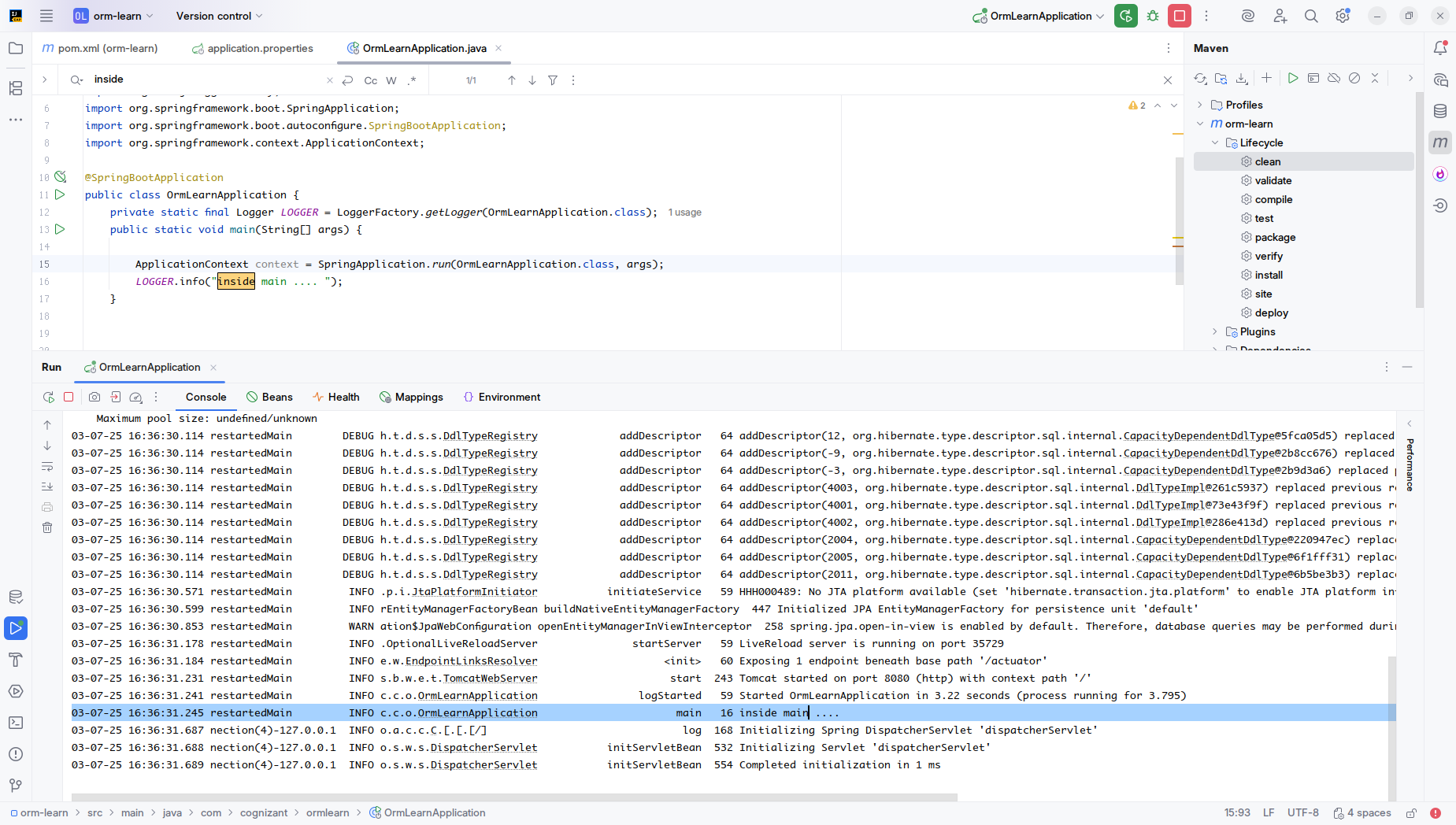
Edited application.properties on src/main/resources/application.properties

Inside intelliJ idea IDE.



Built and cleaned the project using built in maven commands. Logs were included and verified.



Log to verify inside main is shown here: 

SME walkthrough critical points :

## **@SpringBootApplication —**

Equivalent to writing 3 annotations:

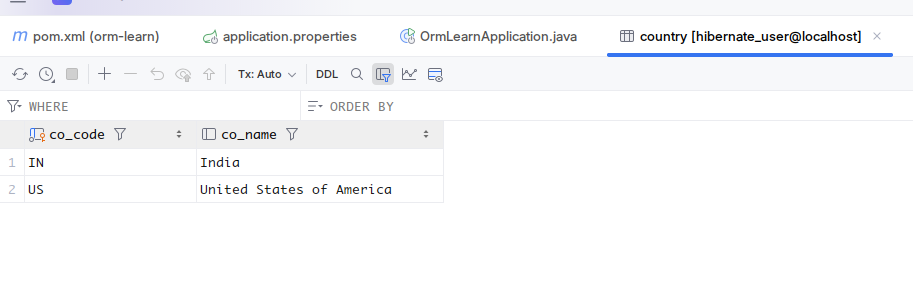
@Configuration // Marks this class as a config class

@EnableAutoConfiguration // Enables Spring Boot's auto configuration

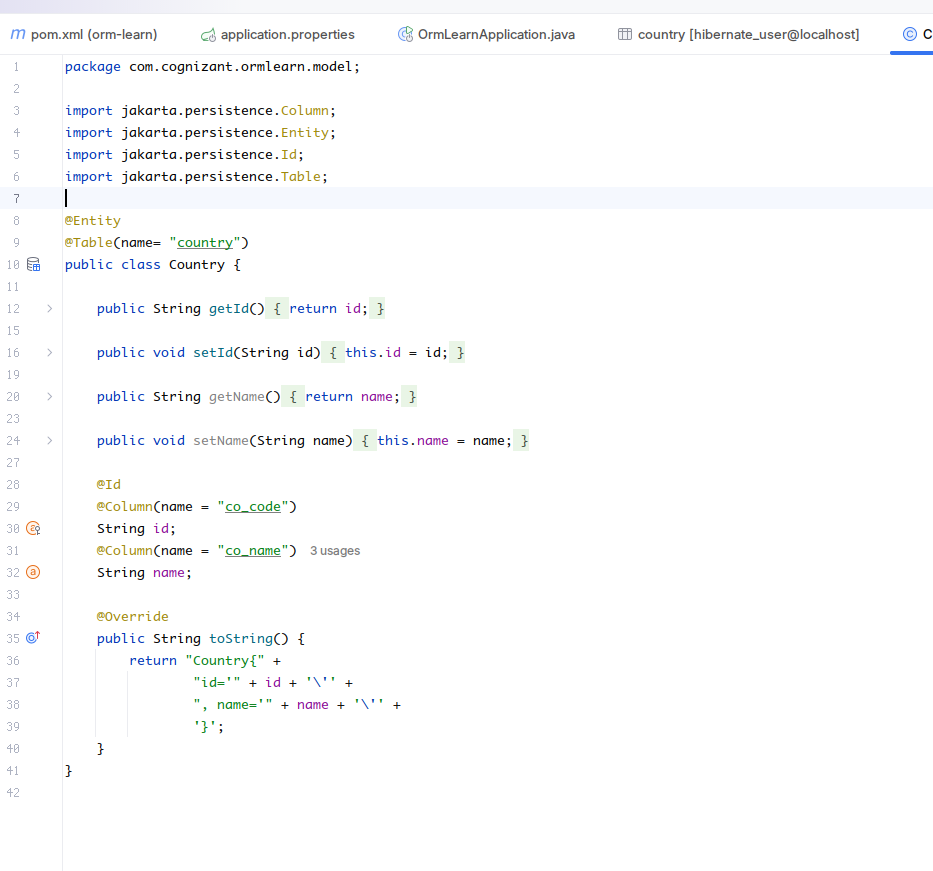
@ComponentScan // Scans the package for @Component/@Service/@Controller

**Country table population:**

The country table was created before and added with new data as shown :



**Persistence Class - com.cognizant.orm-learn.model.Country :**



The persistence class was created and updated with getters and setters along with its own way of “toString(...)” method.

**Repository Class - com.cognizant.orm-learn.CountryRepository**



The repository interface “CountryRepository” was created extending JPA Repository to make use of JPA’s functionality.

Jakarta Persistence API (JPA)

-> It is a standard API for ORM.

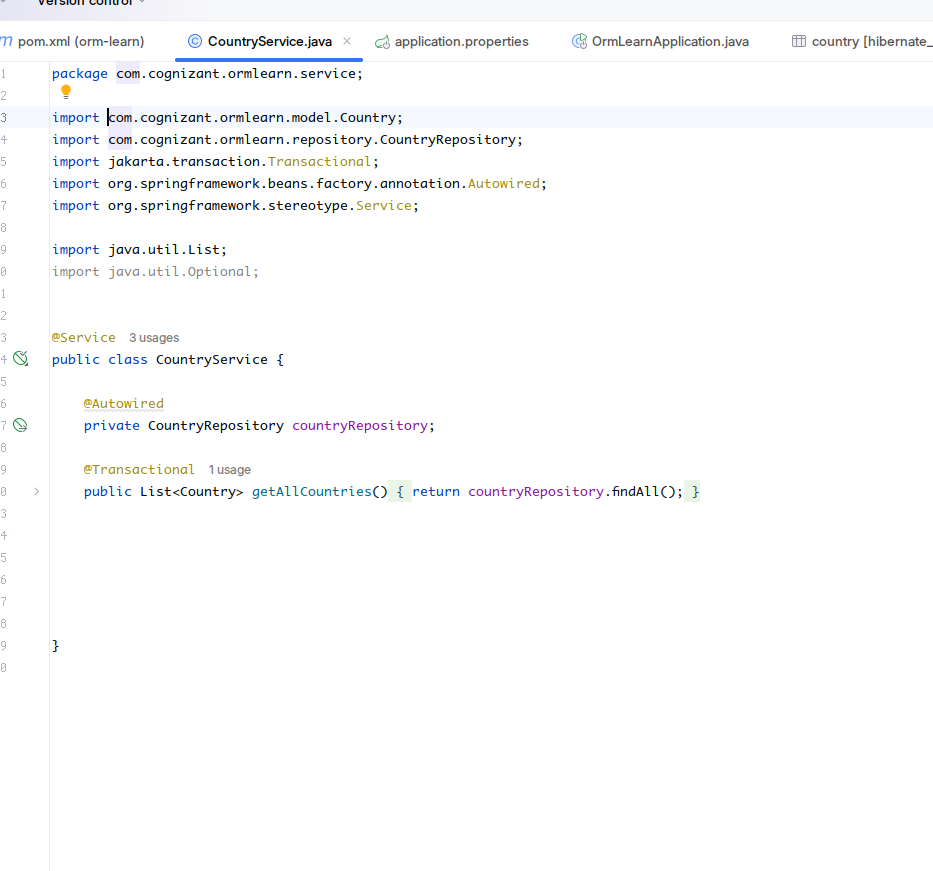
-> It consist only specification

|- define a set of interfaces.

|- It requires implementation.

-> By having this API, we are **not locked to vendor/framework** implementation.

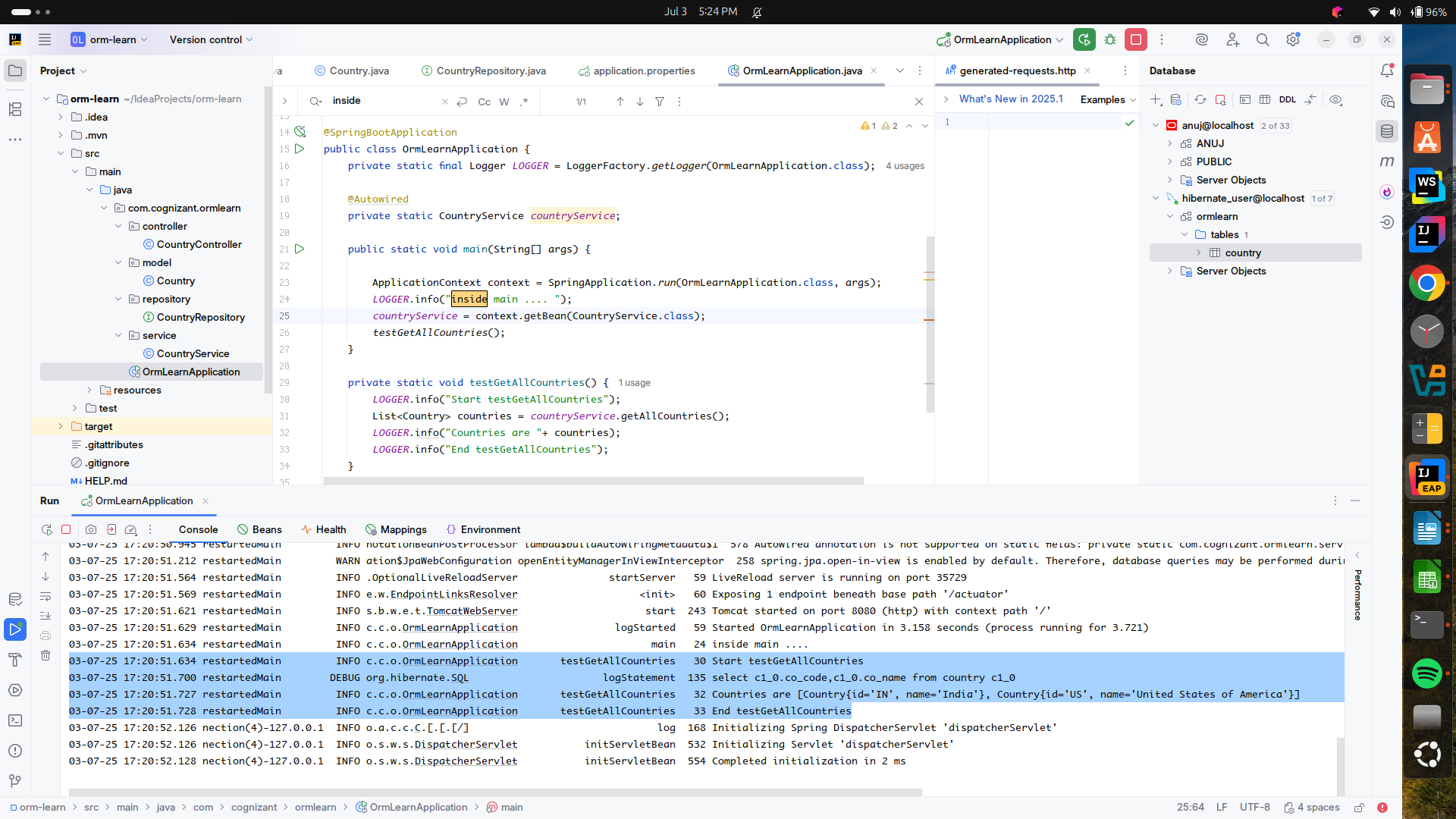
**Service Class - com.cognizant.orm-learn.service.CountryService**

****

The service class “CountryService” was created with @Service annotation autowiring the CountryRepository interface object to it as member variable. Method “getAllCountries(...)” was implemented invoking the jpa’s “findAll()”

**Testing in OrmLearnApplication.java**

Test method was invoked in main class entry point. It tests get All countries method in Services class and

****