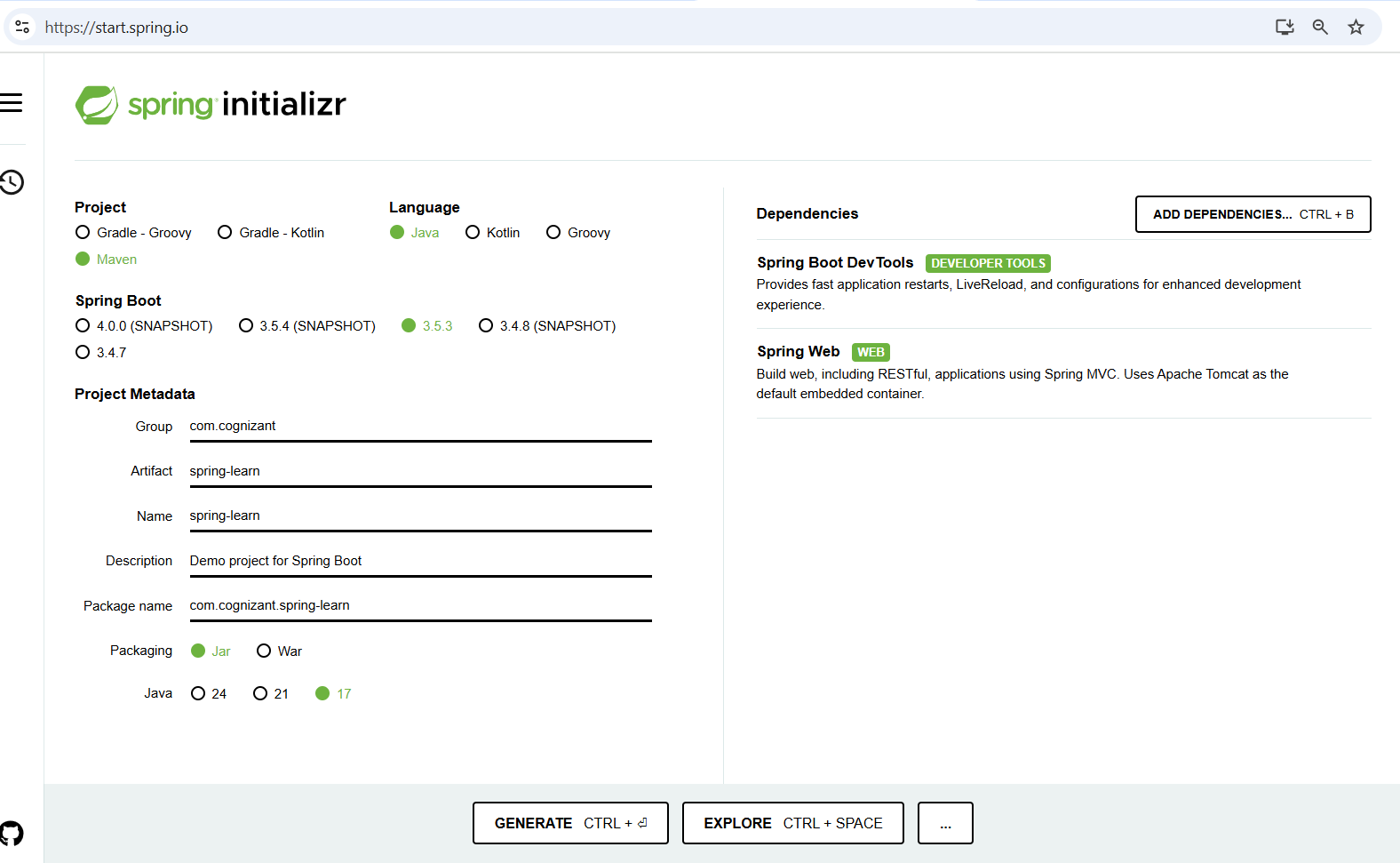
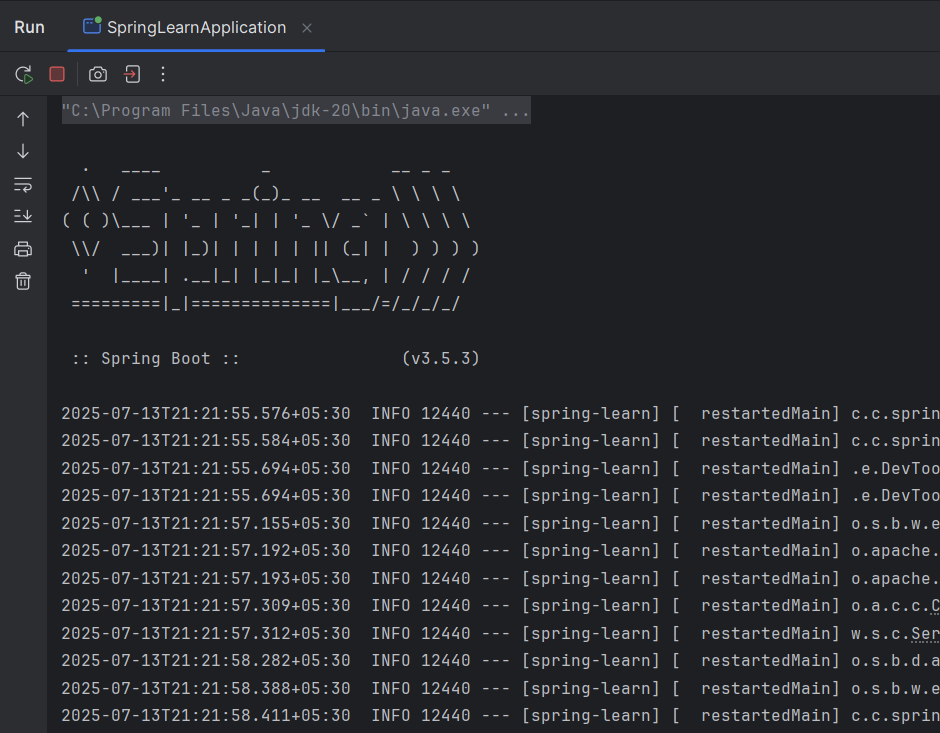
**Cognizant - DN 4.0 Deep Skilling Stage**

**Mandatory Hands-On Week 4 Solutions**

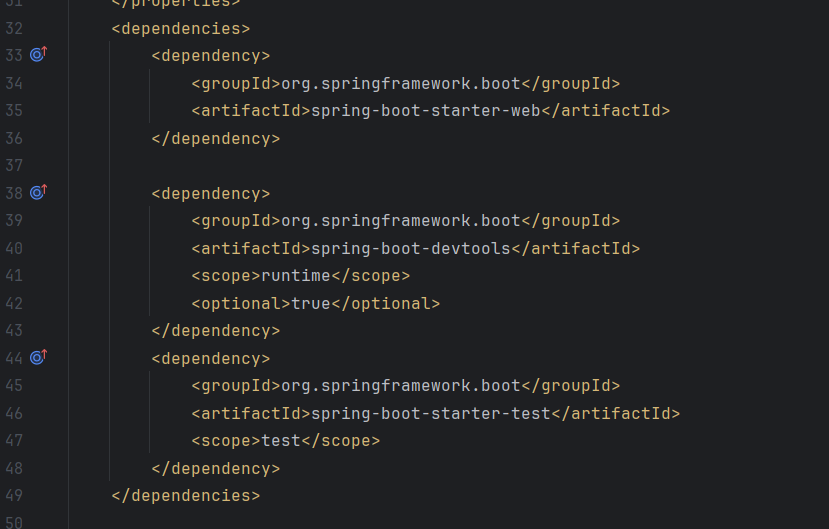
1. **Spring REST Using Spring Boot**

**Exercise 1: Create a Spring Web Project using Maven**

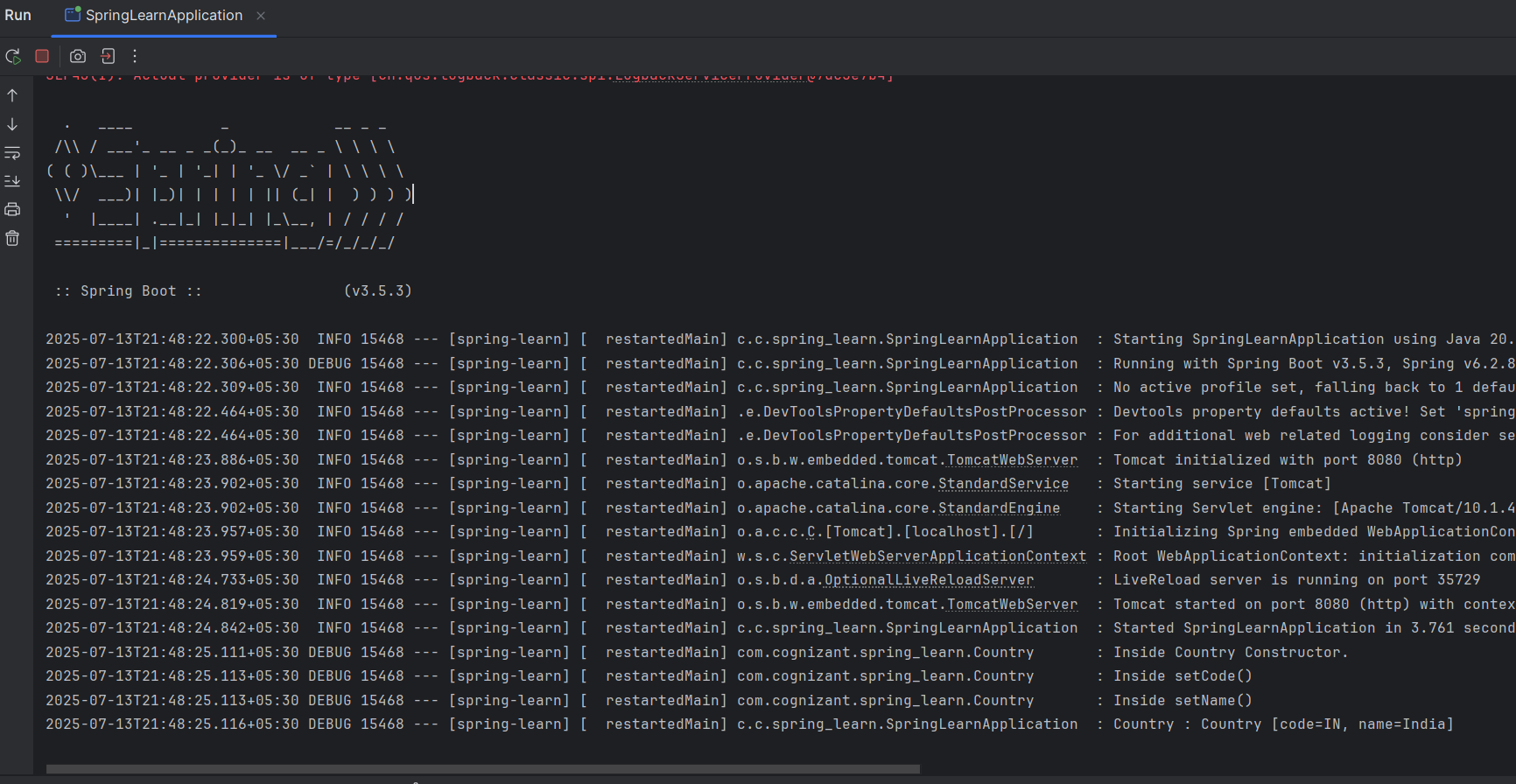
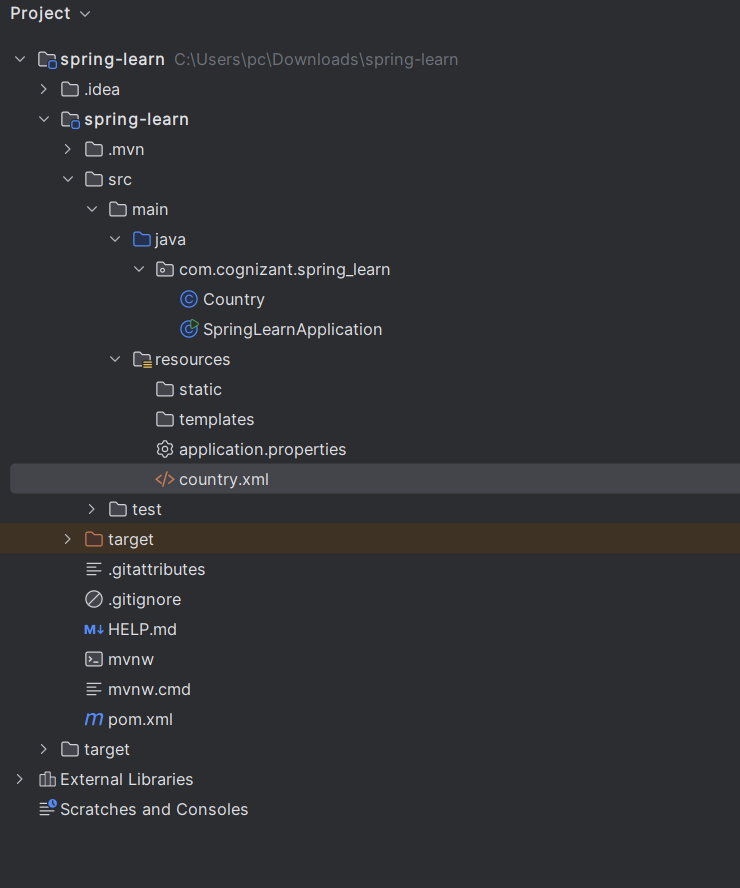
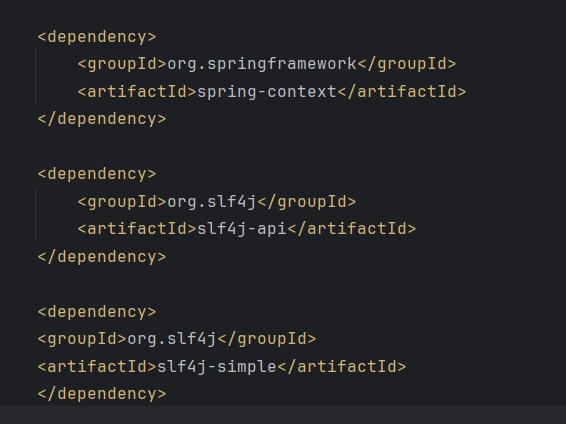
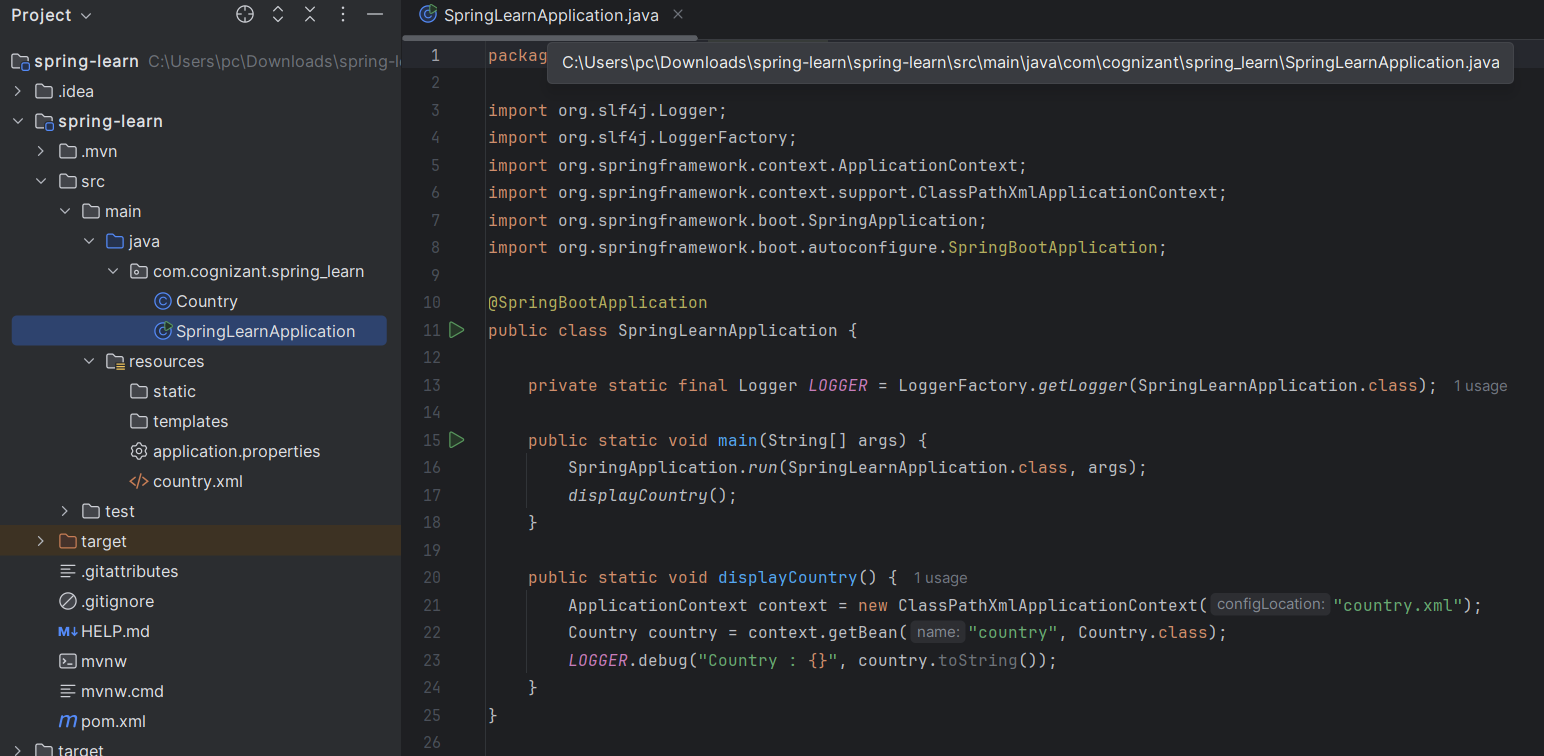
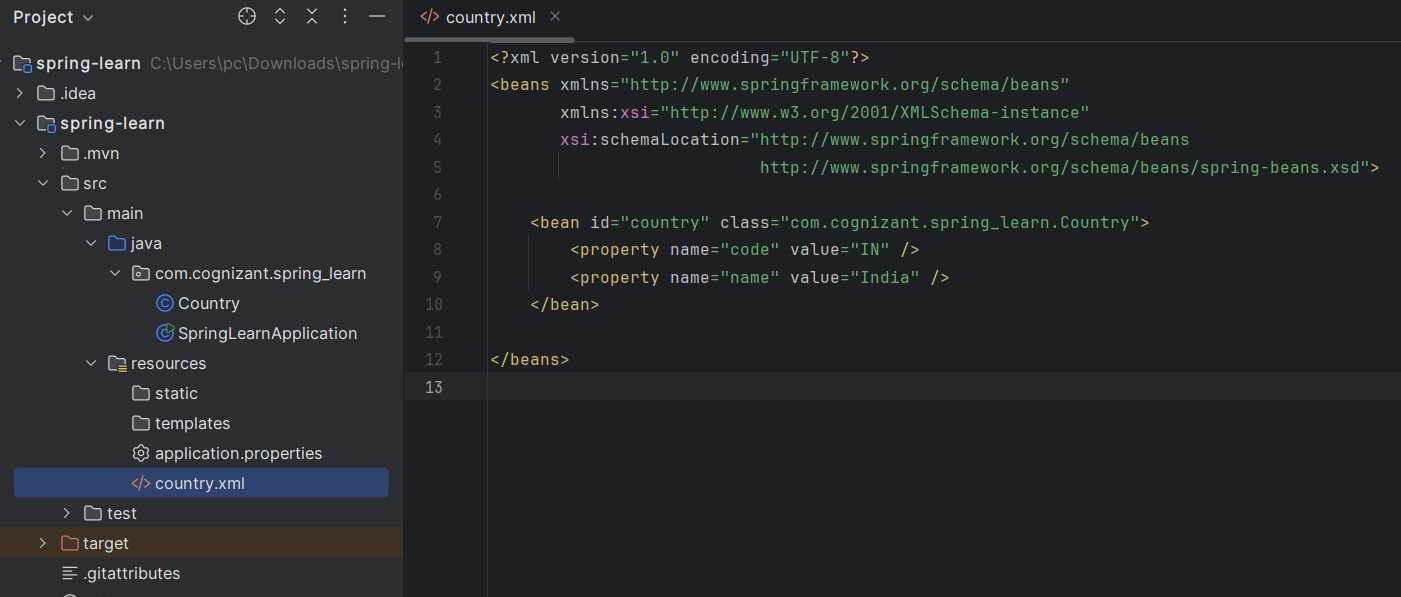
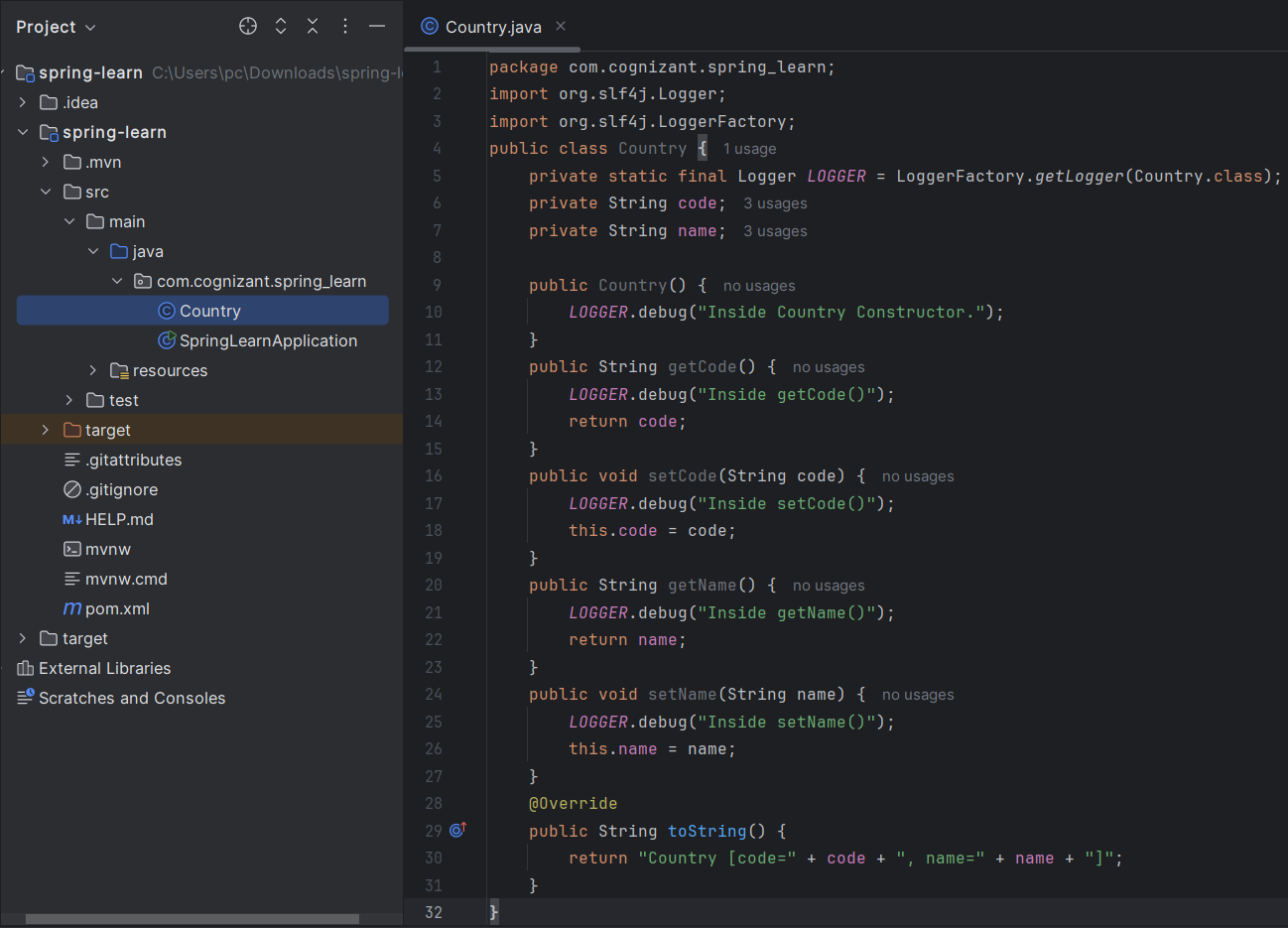
****

****

**pom.xml**

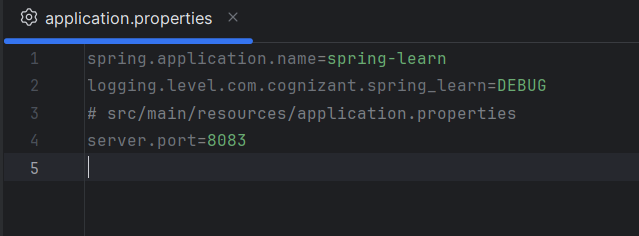
****

**Exercise 2: Spring Core – Load Country from Spring Configuration XML**

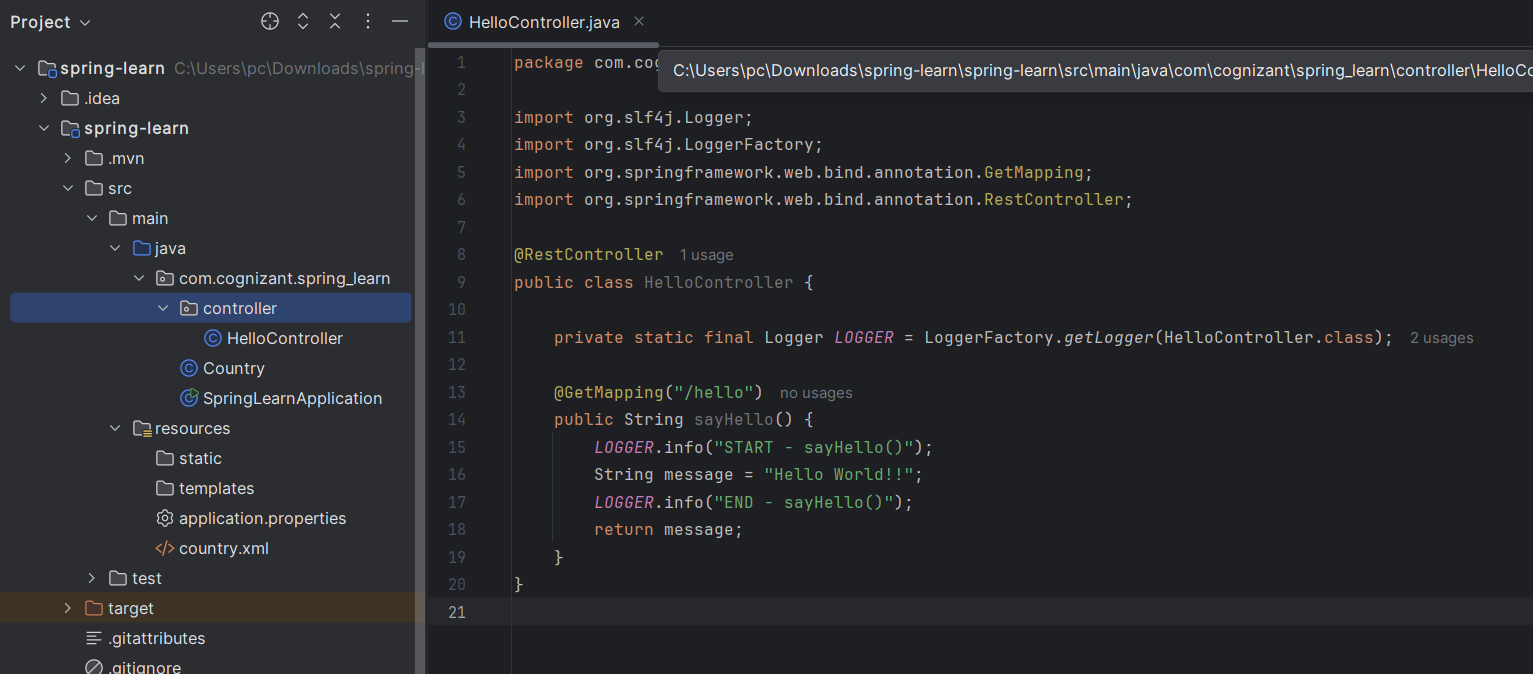
****

**Exercise 3: Spring Core – Load Country from Spring Configuration XML**

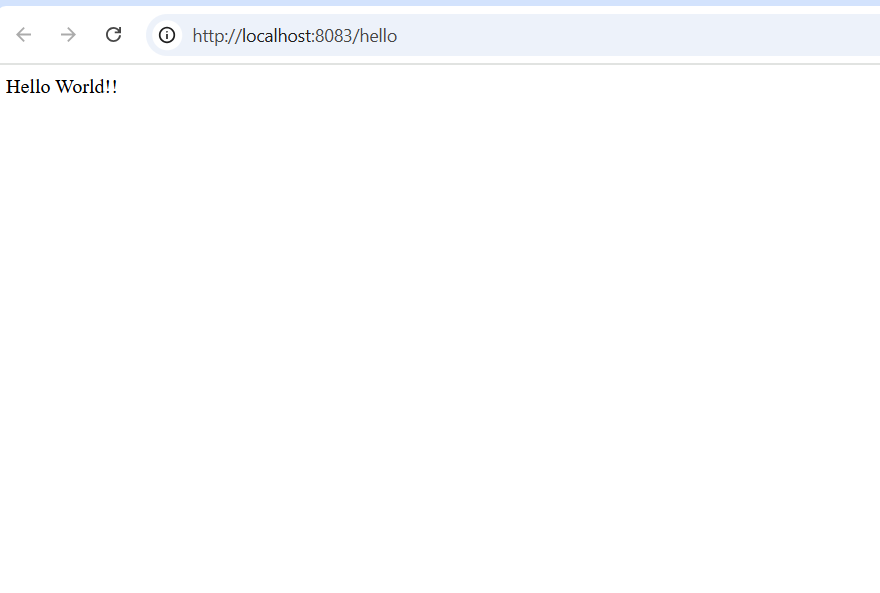
**application.properties**

****

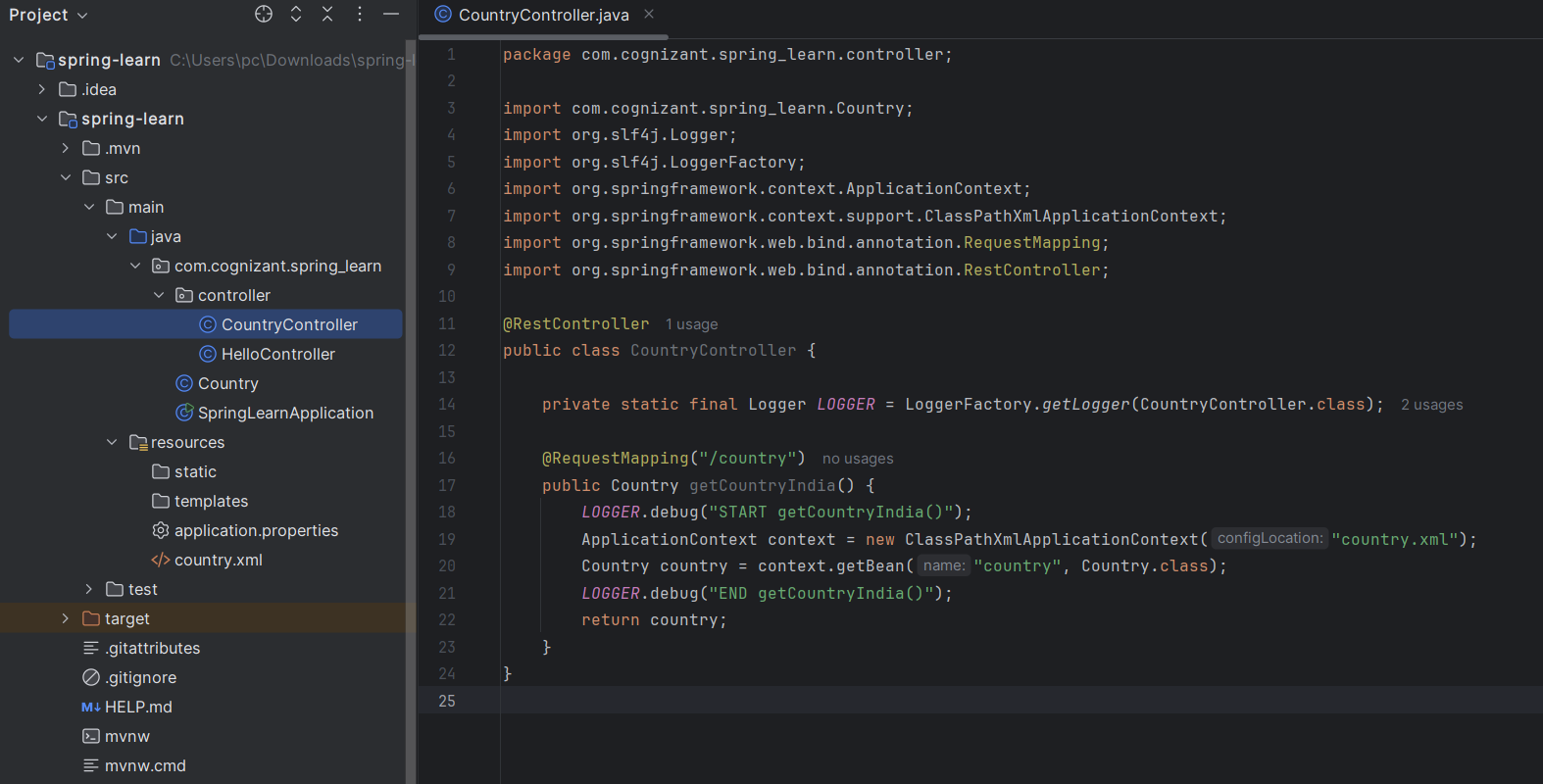
**src/main/java/com/cognizant/spring\_learn/controller/HelloController.java**

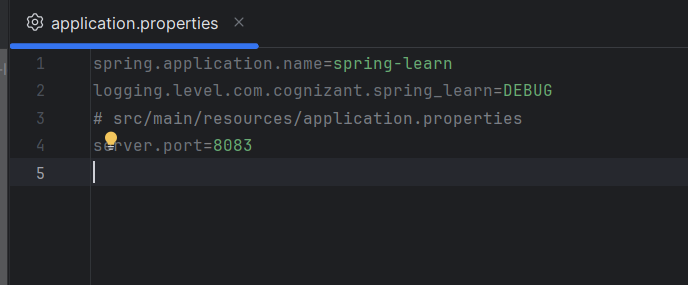
****

**Output:**

****

**Exercise 4: REST - Country Web Service**

****

****

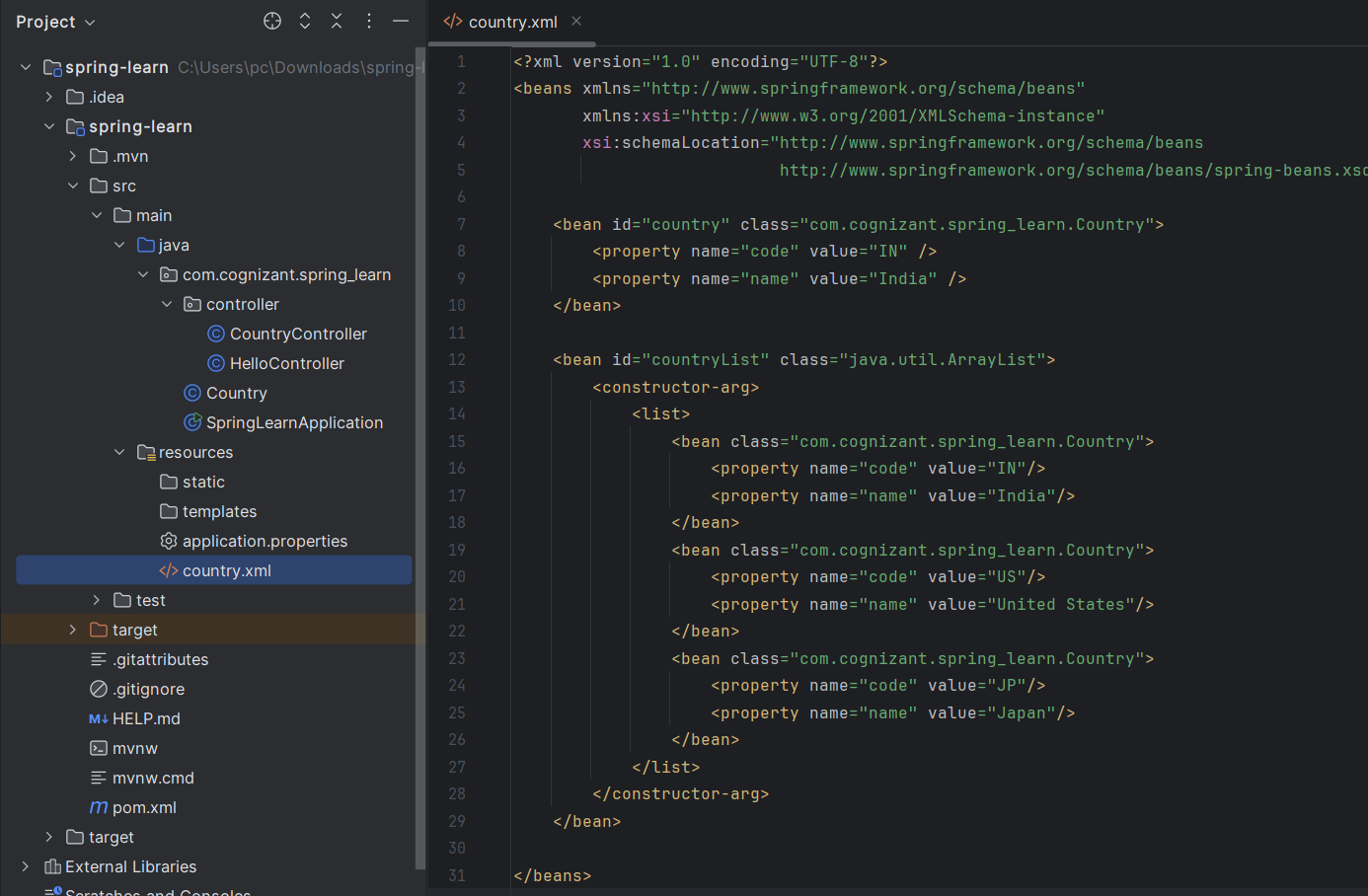
**Output:**

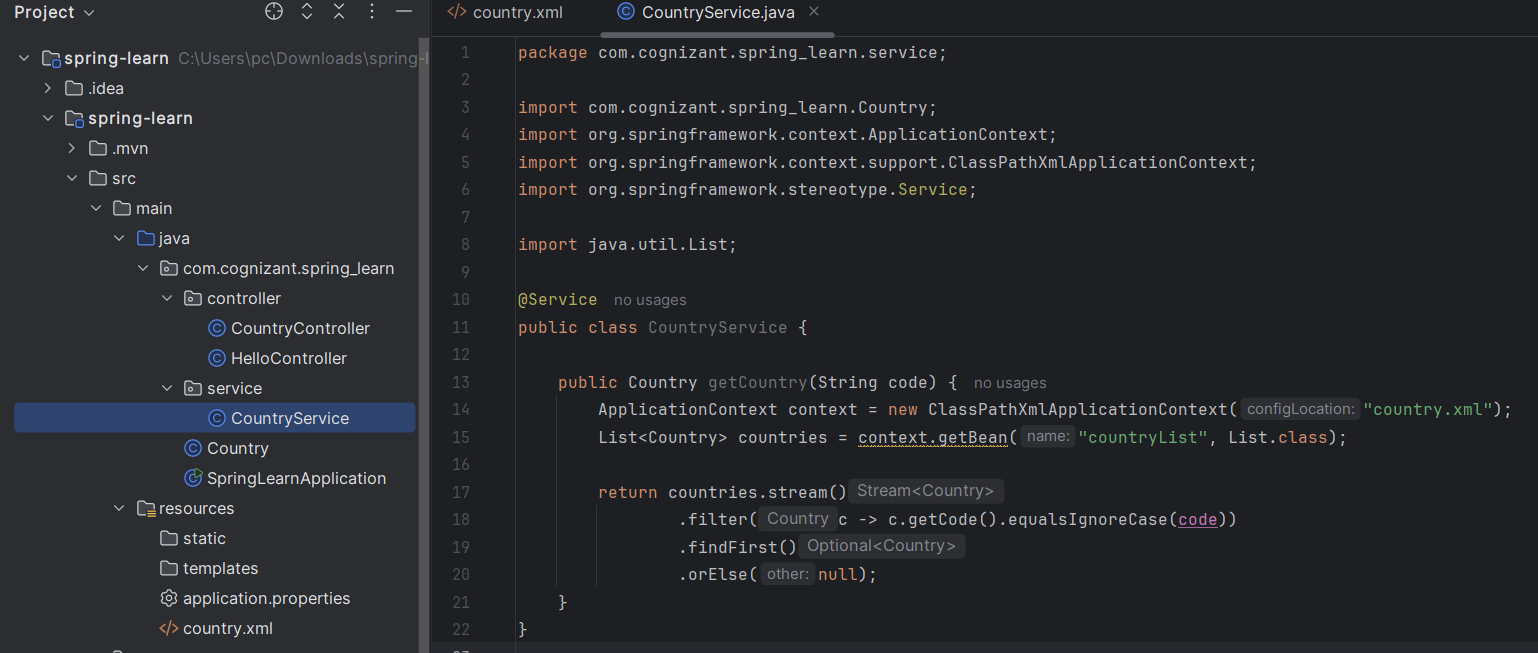
****

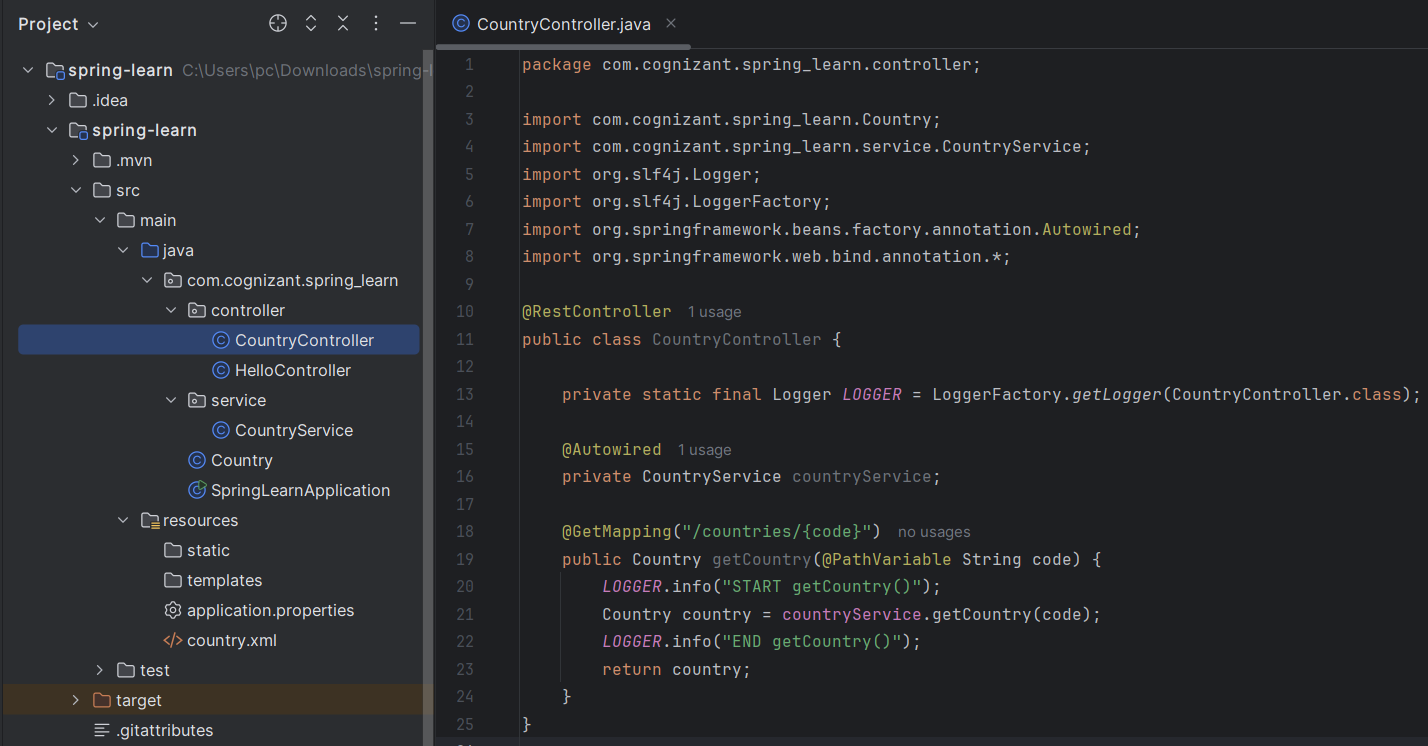
### SME Explanation Points

* What happens in the controller method?  
   Spring calls getCountryIndia(), loads country.xml, fetches the "country" bean, and returns it.
* How is the bean converted to JSON?  
   Spring Boot uses Jackson (from spring-boot-starter-web) to auto-convert the Java object to JSON because @RestController + returning a POJO = JSON by default.
* Where to see HTTP headers?  
  + In Chrome → Right-click → Inspect → Network tab → click the /country request → check Headers tab.
  + In Postman → Send request to /country → click Headers tab to view response headers.

**Exercise 5: REST - Get country based on country code**

****

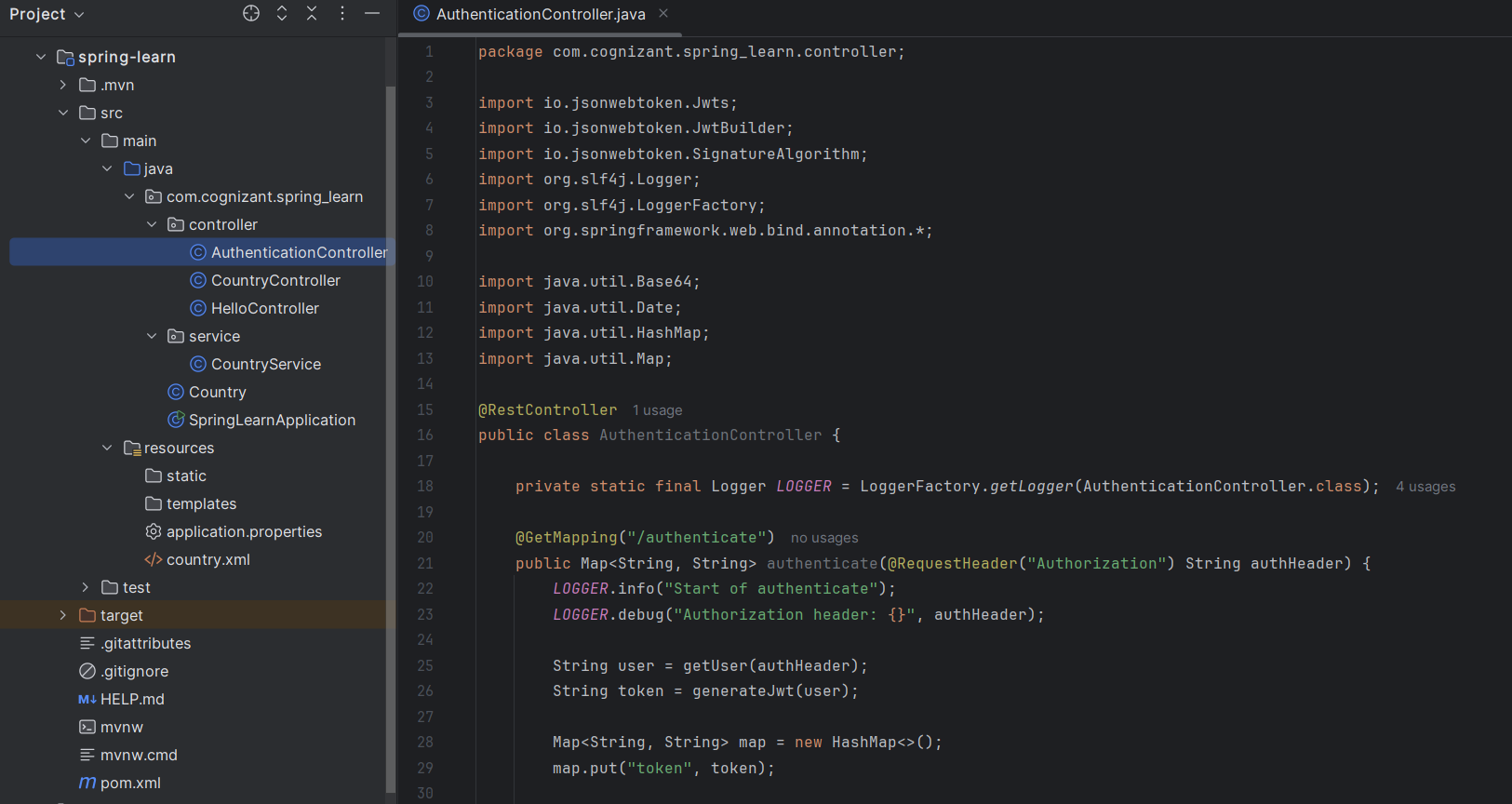
****

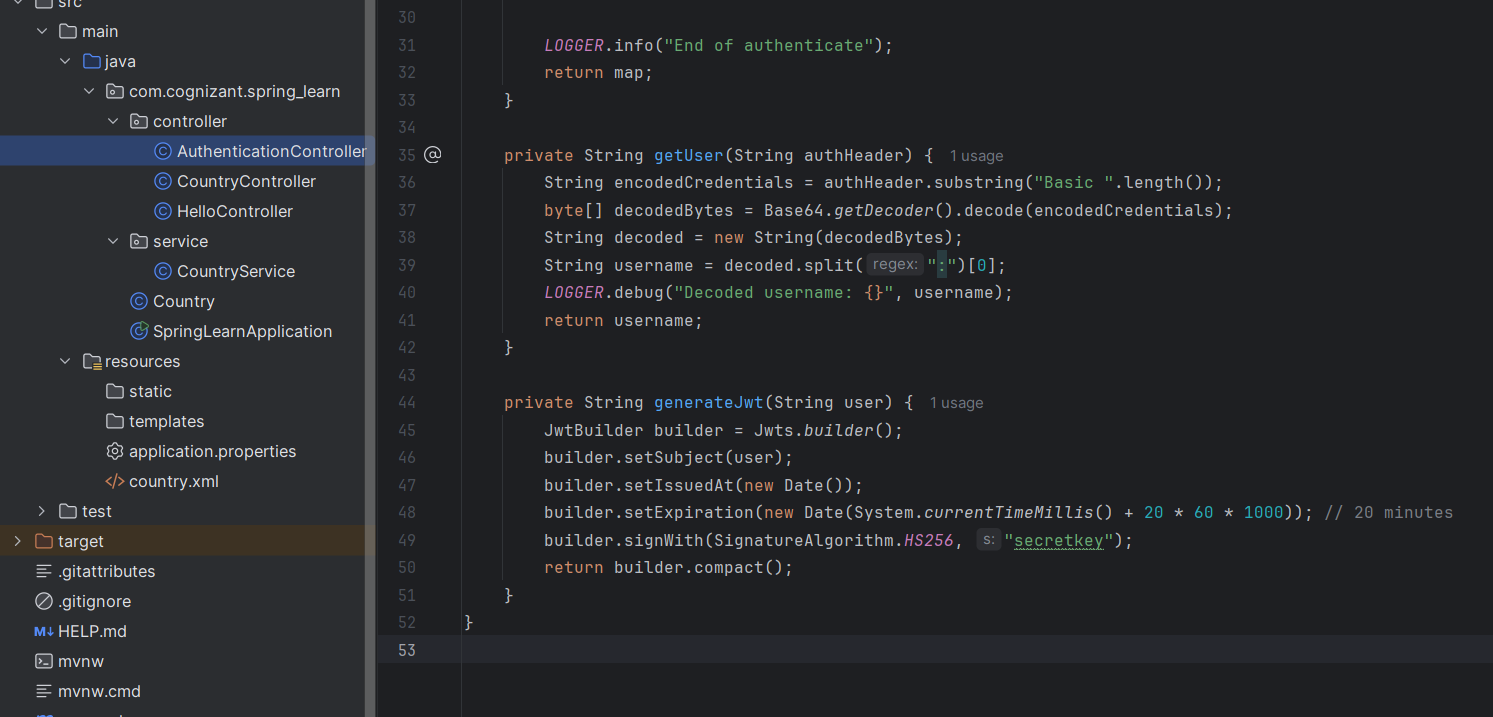
****

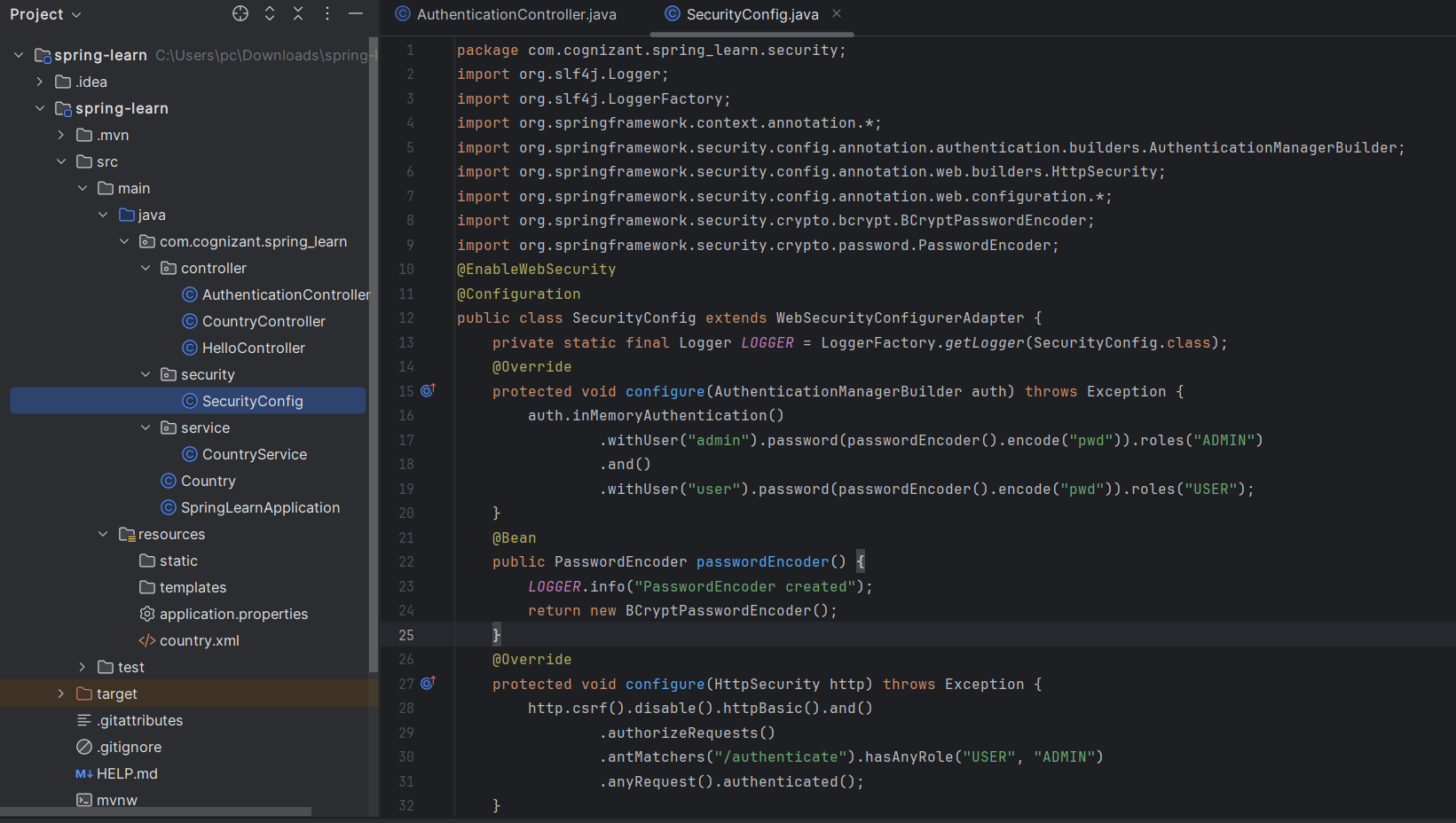
****

**Exercise 6: REST - Create authentication service that returns JWT**

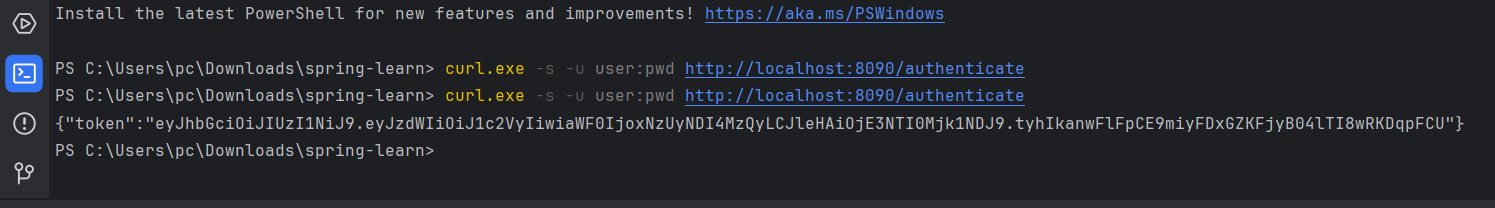
****

****

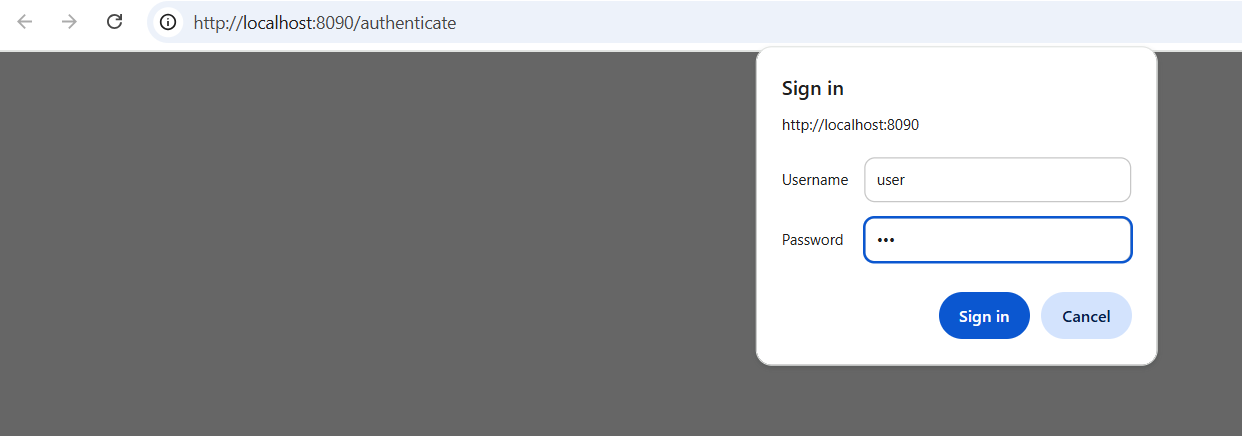
****

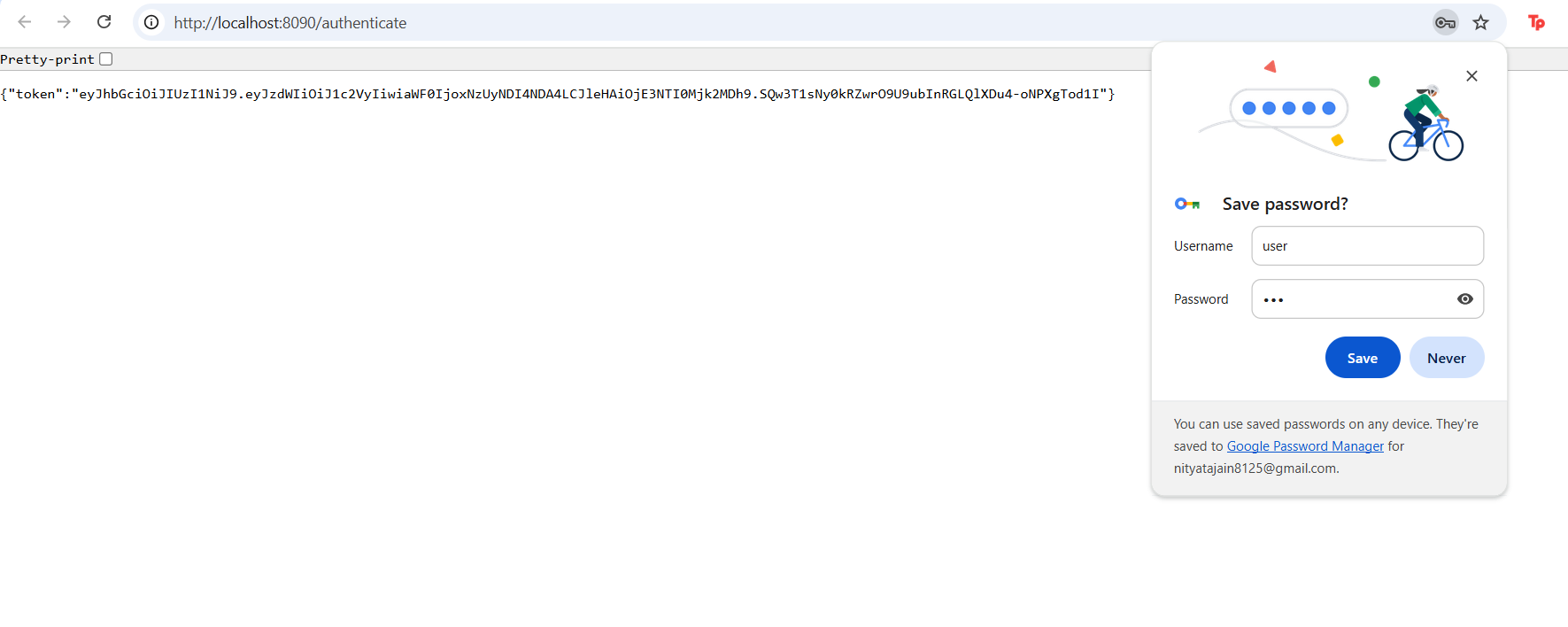
****

**In Terminal:**

****

**Output** [**http://localhost:8090/authenticate**](http://localhost:8090/authenticate)

****

****