**WEEK 4**

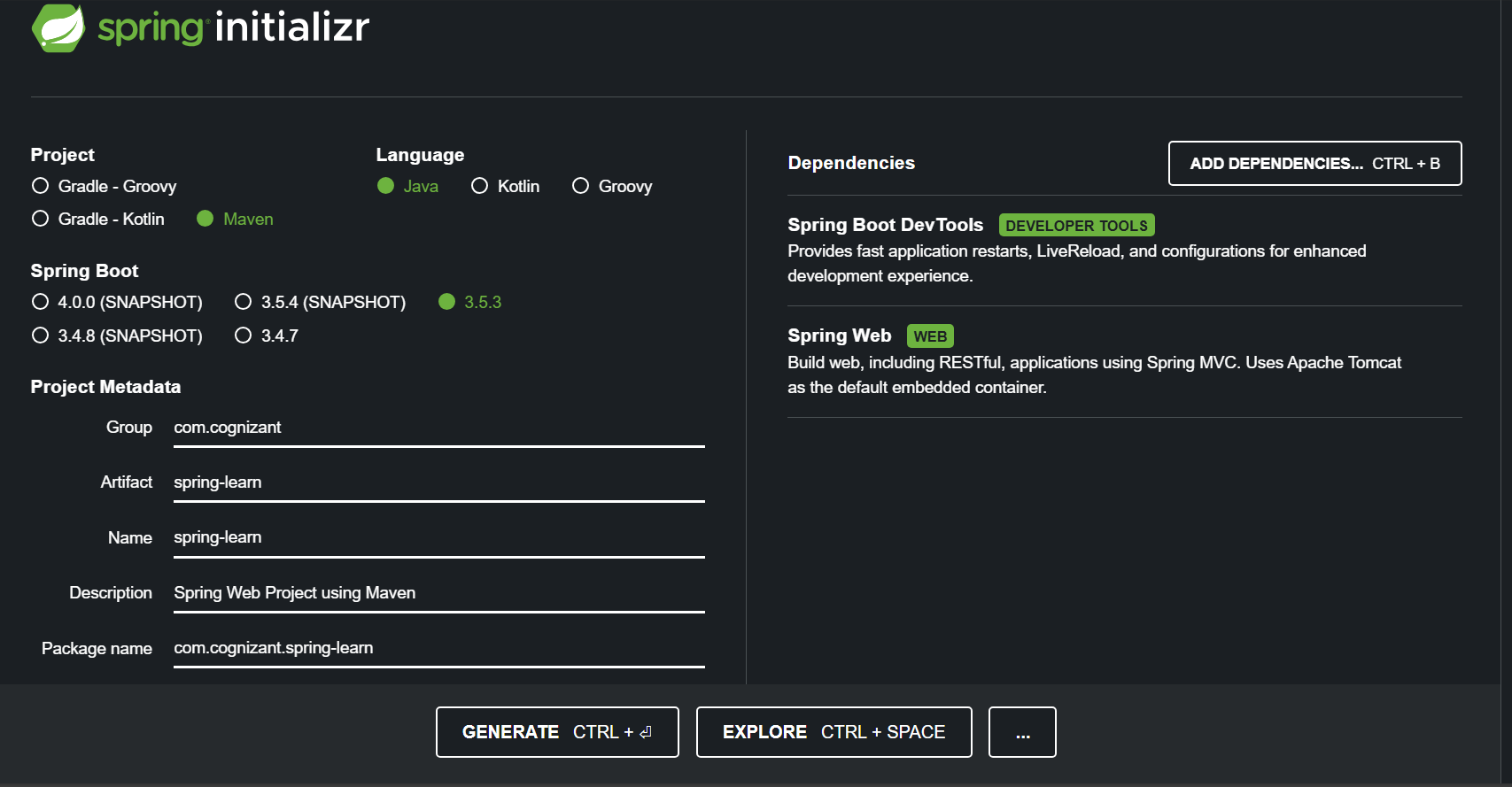
**SPRING REST USING SPRING BOOT**

**1. Spring Rest HandsOn.docx**

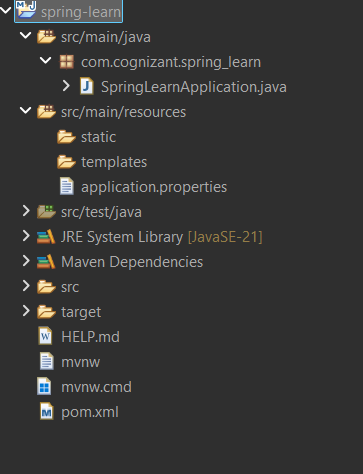
**Hands on 1**

**Create a Spring Web Project using Maven**

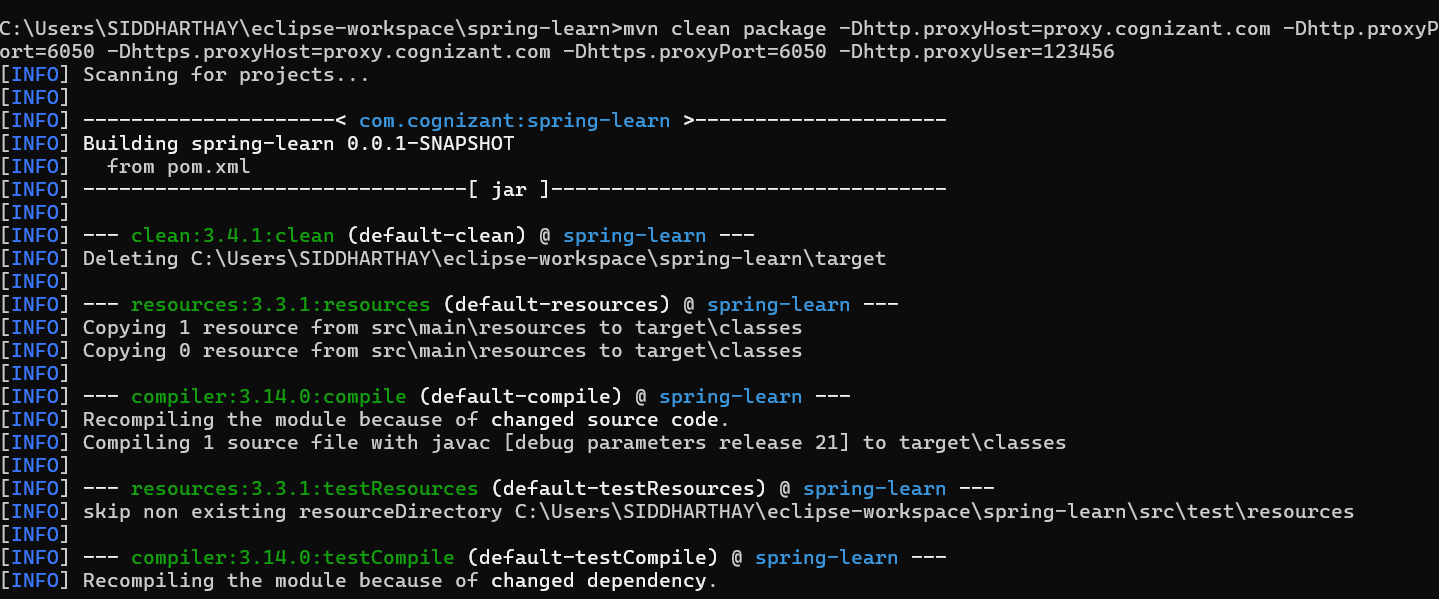
**1. Creating the Spring Web Project using Spring Initializr:**

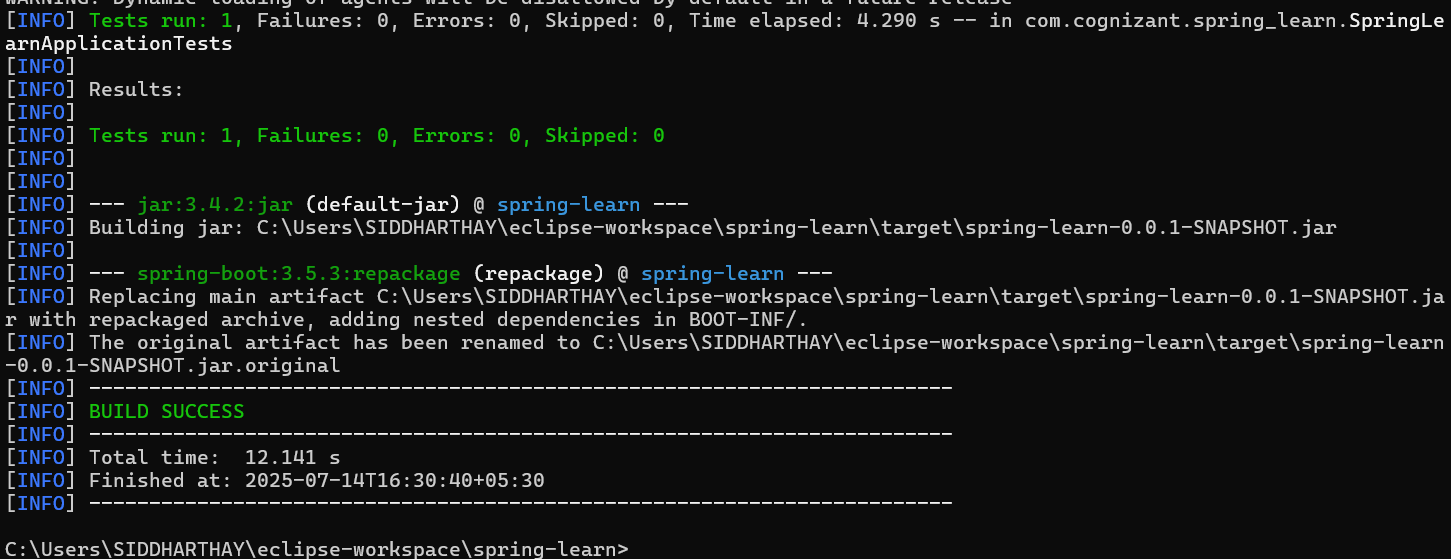


**2. Project Structure after Importing it into Eclipse:**

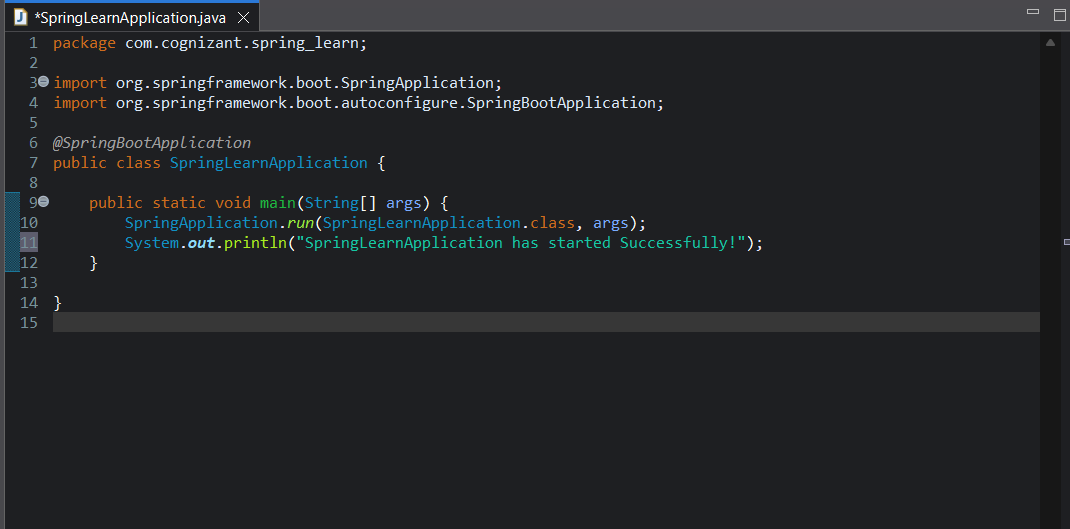


**3. Building the Project using Maven(proxy):**

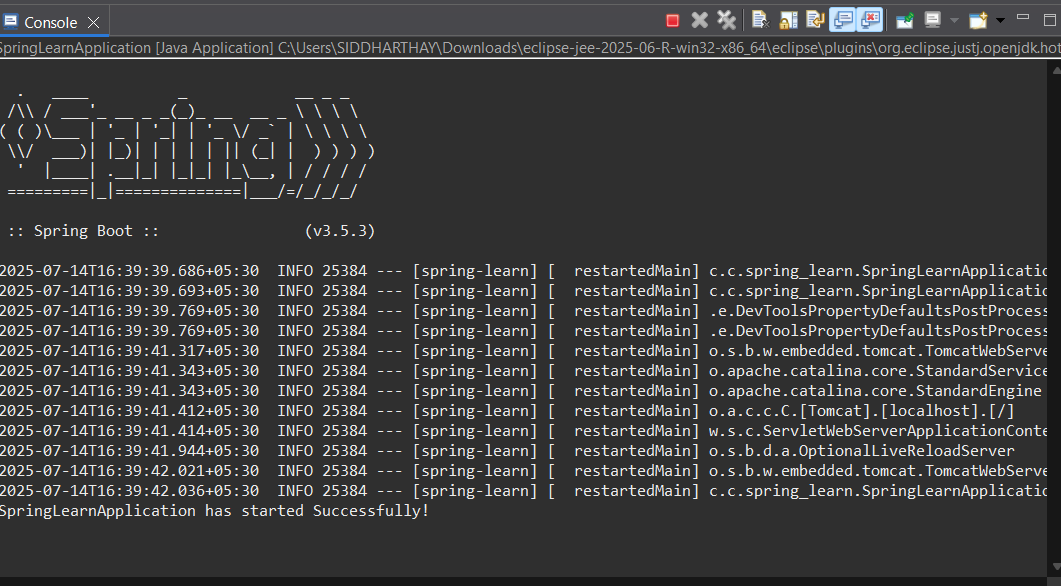




**4. Adding Logging in Main() method:**



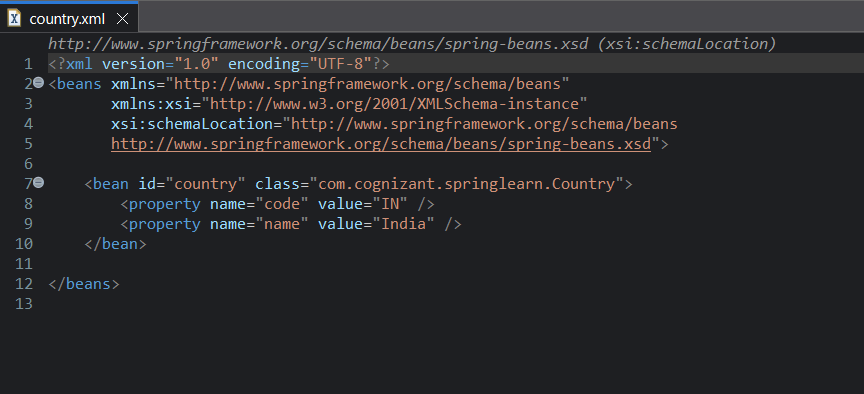
**5. Output:**



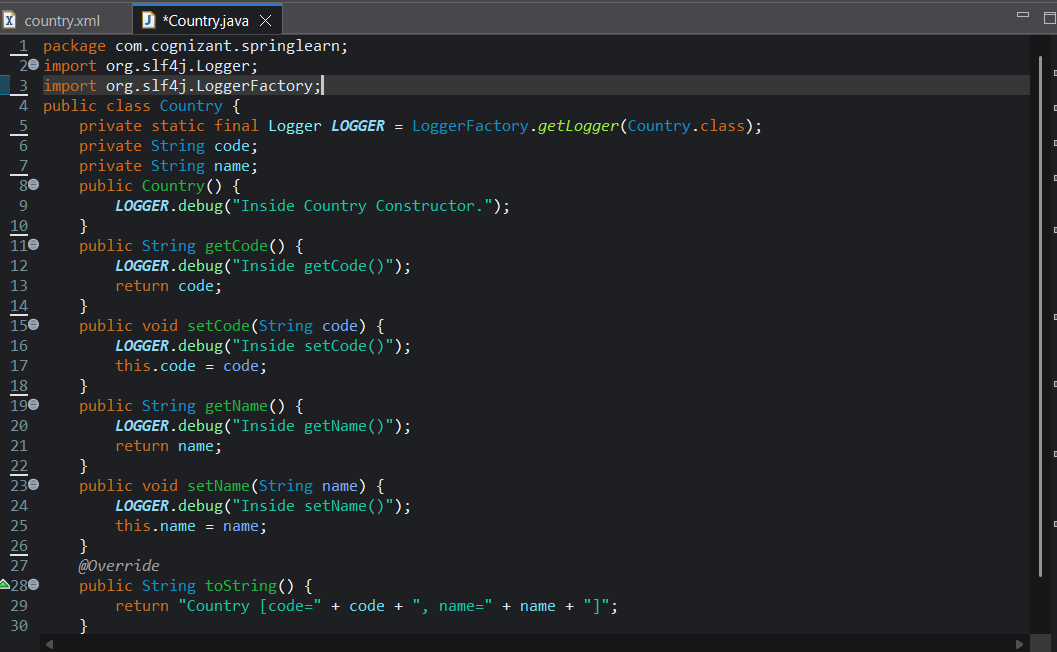
**Hands on 4**

**Spring Core – Load Country from Spring Configuration XML** 

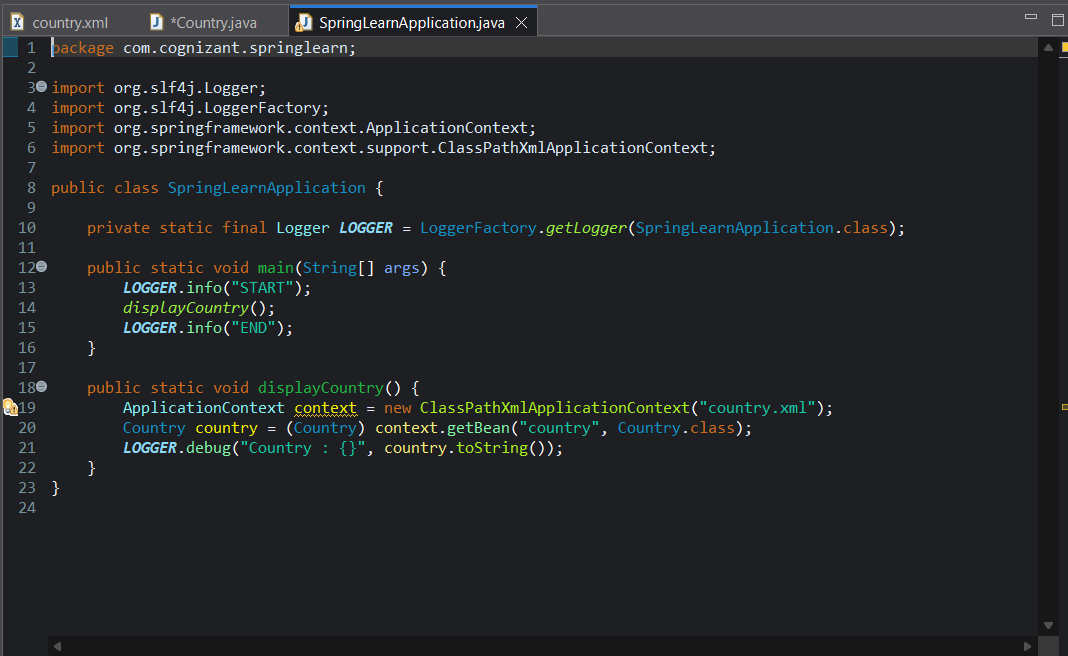
**1. country.xml config file bean definition of country class:**



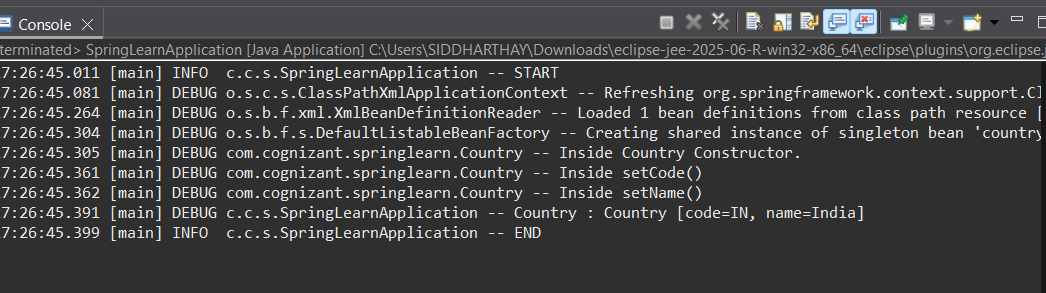
**2. Country.java shows all getter and setter methods**



**3. Main class shows displayCountry() method and how bean is loaded:**



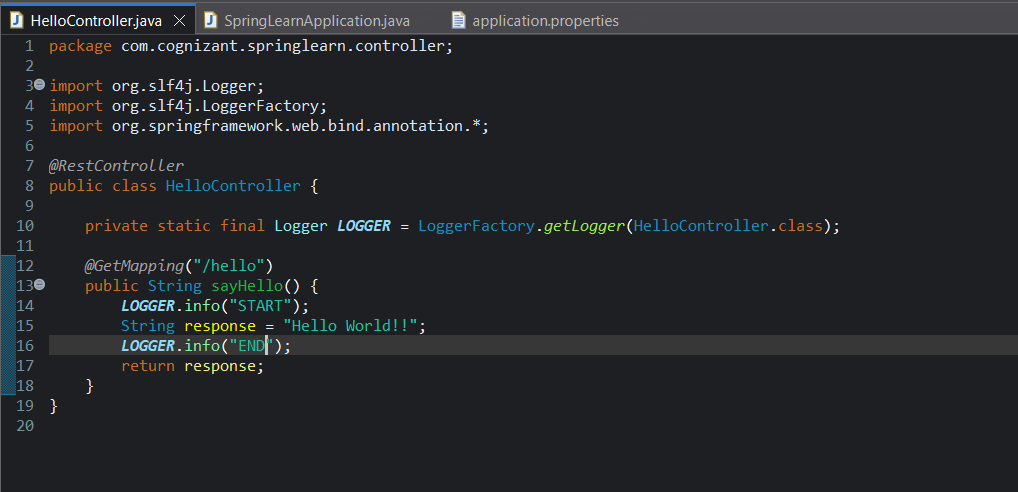
**4. Console showing debug logs:**



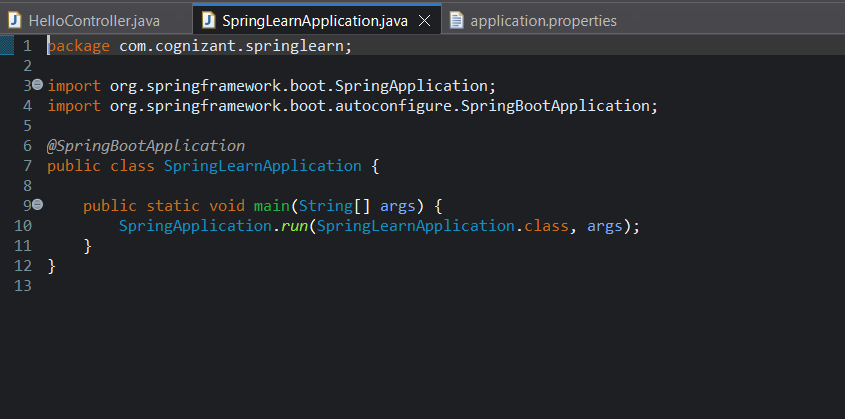
**2. Spring Rest HandsOn.docx**

**Hello World RESTful Web Service**   
  
Write a REST service in the spring learn application created earlier, that returns the text "Hello World!!" using Spring Web Framework. Refer details below:  
  
**Method:** GET  
**URL:** /hello  
**Controller:** com.cognizant.spring-learn.controller.HelloController  
**Method Signature:** public String sayHello()  
**Method Implementation:** return hard coded string "Hello World!!"  
**Sample Request**: http://localhost:8083/hello  
**Sample Response:** Hello World!!

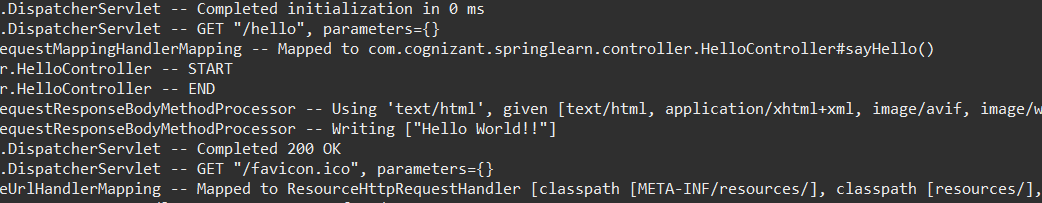
**1. HelloController.java Rest Controller:**



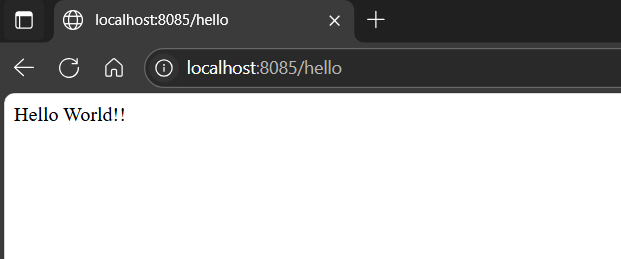
**2. Main() for running the application:**



**3. Console:**



**4. Response through web browser:**



**5. Response through Postman:**

