

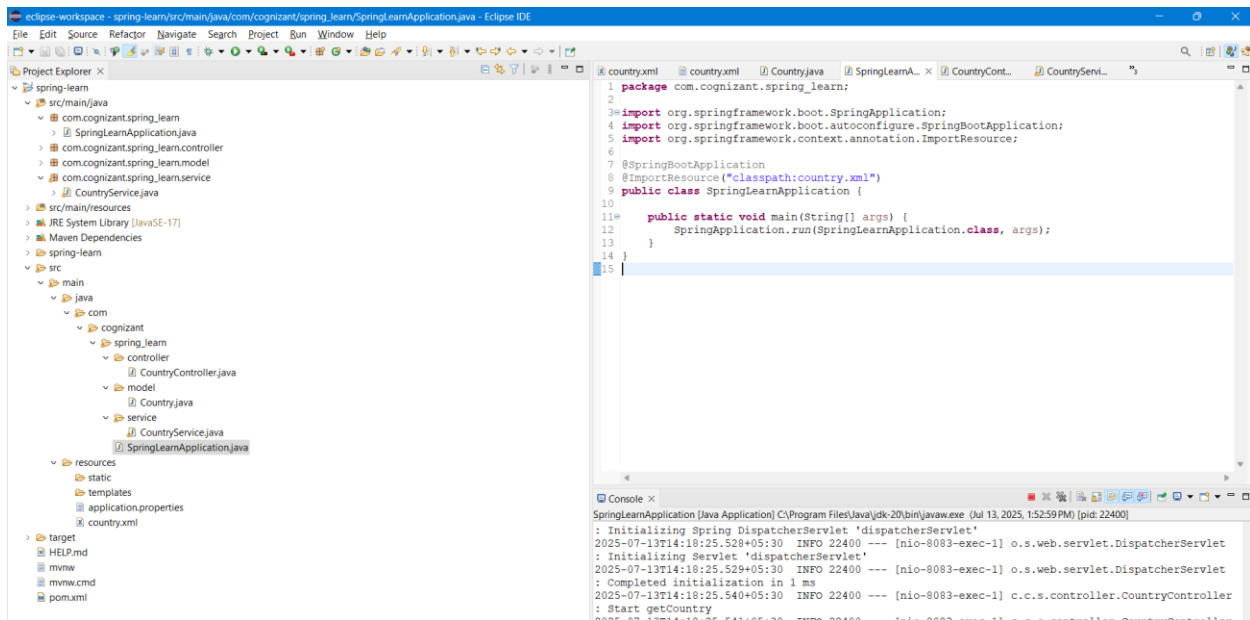
Digital Nurture 4.0 Deep Skilling

Week 4 – HANDS ON 5

5.REST - Get country based on country code

Write a REST service that returns a specific country based on country code. The country code should be case insensitive.

Code :



The screenshot shows the Eclipse IDE with the SpringLearnApplication.java file open. The code defines a Spring Boot application with a main method that runs the application. The console output shows the application starting successfully, with logs for the Spring DispatcherServlet, Initializing Servlet, and Completed initialization.

```
1 package com.cognizant.spring_learn;
2
3 import org.springframework.boot.SpringApplication;
4 import org.springframework.boot.autoconfigure.SpringBootApplication;
5 import org.springframework.context.annotation.ImportResource;
6
7 @SpringBootApplication
8 @ImportResource("classpath:country.xml")
9 public class SpringLearnApplication {
10
11     public static void main(String[] args) {
12         SpringApplication.run(SpringLearnApplication.class, args);
13     }
14 }
15
```

```
SpringLearnApplication [Java Application] C:\Program Files\Java\jdk-20\bin\java.exe (Jul 13, 2025, 1:52:59 PM) [pid: 22400]
: Initializing Spring DispatcherServlet 'dispatcherServlet'
2025-07-13T14:18:25.529+05:30 INFO 22400 --- [nio-8083-exec-1] o.s.web.servlet.DispatcherServlet
: Initializing Servlet 'dispatcherServlet'
2025-07-13T14:18:25.529+05:30 INFO 22400 --- [nio-8083-exec-1] o.s.web.servlet.DispatcherServlet
: Completed initialization in 1 ms
2025-07-13T14:18:25.540+05:30 INFO 22400 --- [nio-8083-exec-1] c.c.s.controller.CountryController
: Start getCountry
2025-07-13T14:18:25.540+05:30 INFO 22400 --- [nio-8083-exec-1] c.c.s.controller.CountryController
```

Country.xml



The screenshot shows the country.xml file in the Eclipse IDE. The XML defines two beans: a Country bean with code 'IN' and name 'India', and a countryList bean which is an ArrayList containing the Country bean.

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <beans xmlns="http://www.springframework.org/schema/beans"
3       xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
4       xsi:schemaLocation="http://www.springframework.org/schema/beans
5       http://www.springframework.org/schema/beans/spring-beans.xsd">
6
7     <bean id="in" class="com.cognizant.spring_learn.model.Country">
8         <property name="code" value="IN"/>
9         <property name="name" value="India"/>
10    </bean>
11
12    <bean id="countryList" class="java.util.ArrayList">
13        <constructor-arg>
14            <list>
15                <ref bean="in"/>
16                <!-- Add more countries as needed -->
17            </list>
18        </constructor-arg>
19    </bean>
20
21 </beans>
```

Country.java

```
country.xml  country.xml  *Country.java x  SpringLearnA...  CountryCont...  CountrySe
1 package com.cognizant.spring_learn.model;
2 public class Country {
3     private String code;
4     private String name;
5
6     public Country() {}
7
8     public Country(String code, String name) {
9         this.code = code;
10        this.name = name;
11    }
12    public String getCode() {
13        return code;
14    }
15    public void setCode(String code) {
16        this.code = code;
17    }
18    public String getName() {
19        return name;
20    }
21    public void setName(String name) {
22        this.name = name;
23    }
24    @Override
25    public String toString() {
26        return "Country [code=" + code + ", name=" + name + "]";
27    }
28 }
29
```

CountryService.java

```
country.xml  country.xml  *Country.java  SpringLearnA...  CountryCont...  CountryServi... x  "3
1 package com.cognizant.spring_learn.service;
2
3 import com.cognizant.spring_learn.model.Country;
4 import org.springframework.beans.factory.annotation.Autowired;
5 import org.springframework.context.ApplicationContext;
6 import org.springframework.stereotype.Service;
7
8 import java.util.List;
9
10 @Service
11 public class CountryService {
12
13     @Autowired
14     private ApplicationContext context;
15
16     public Country getCountry(String code) throws Exception {
17         List<Country> countries = (List<Country>) context.getBean("countryList");
18
19         return countries.stream()
20             .filter(c -> c.getCode().equalsIgnoreCase(code))
21             .findFirst()
22             .orElseThrow(() -> new Exception("Country not found with code: " + code));
23     }
24 }
```

Port : 8083 with header



Postman :

