Spring REST using Spring Boot 3(1. spring-rest-handson)

1.Create a Spring Web Project using Maven

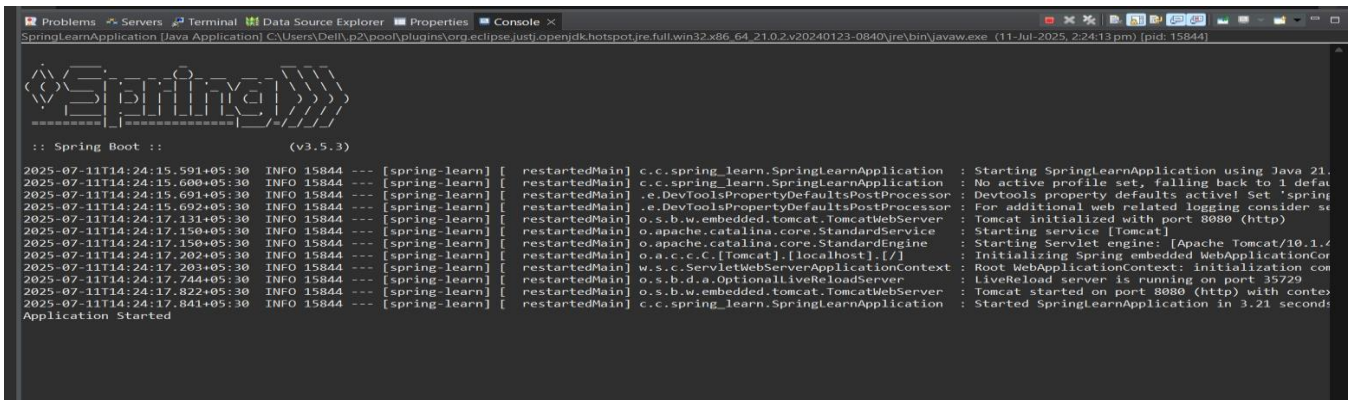
Follow steps below to create a project:

1. Go to https://start.spring.io/
2. Change Group as “com.cognizant”
3. Change Artifact Id as “spring-learn”
4. Select Spring Boot DevTools and Spring Web
5. Create and download the project as zip
6. Extract the zip in root folder to Eclipse Workspace
7. Build the project using ‘mvn clean package -Dhttp.proxyHost=proxy.cognizant.com -Dhttp.proxyPort=6050 -Dhttps.proxyHost=proxy.cognizant.com -Dhttps.proxyPort=6050 -Dhttp.proxyUser=123456’ command in command line
8. Import the project in Eclipse "File > Import > Maven > Existing Maven Projects > Click Browse and select extracted folder > Finish"
9. Include logs to verify if main() method of SpringLearnApplication.
10. Run the SpringLearnApplication class.

SME to walk through the following aspects related to the project created:

1. src/main/java - Folder with application code
2. src/main/resources - Folder for application configuration
3. src/test/java - Folder with code for testing the application
4. SpringLearnApplication.java - Walkthrough the main() method.
5. Purpose of @SpringBootApplication annotation
6. pom.xml
    1. Walkthrough all the configuration defined in XML file
    2. Open 'Dependency Hierarchy' and show the dependency tree.

OUTPUT:



2.Spring Core – Load Country from Spring Configuration XML

An airlines website is going to support booking on four countries. There will be a drop down on the home page of this website to select the respective country. It is also important to store the two-character ISO code of each country.

Code	Name
US	United States
DE	Germany
IN	India
JP	Japan

Above data has to be stored in spring configuration file. Write a program to read this configuration file and display the details.

Steps to implement

- Pick any one of your choice country to configure in Spring XML configuration named country.xml.
- Create a bean tag in spring configuration for country and set the property and values

```
<bean id="country" class="com.cognizant.springlearn.Country">
    <property name="code" value="IN" />
    <property name="name" value="India" />
</bean>
```

- Create Country class with following aspects:
  - Instance variables for code and name
  - Implement empty parameter constructor with inclusion of debug log within the constructor with log message as “Inside Country Constructor.”
  - Generate getters and setters with inclusion of debug with relevant message within each setter and getter method.
  - Generate toString() method
- Create a method displayCountry() in SpringLearnApplication.java, which will read the country bean from spring configuration file and display the country details. ClassPathXmlApplicationContext, ApplicationContext and context.getBean(“beanId”, Country.class). Refer sample code for displayCountry() method below.

```
ApplicationContext context = new ClassPathXmlApplicationContext("country.xml
");

Country country = (Country) context.getBean("country", Country.class);

LOGGER.debug("Country : {}", country.toString());
```

- Invoke displayCountry() method in main() method of SpringLearnApplication.java.
- Execute main() method and check the logs to find out which constructors and methods were invoked.

SME to provide more detailing about the following aspects:

- bean tag, id attribute, class attribute, property tag, name attribute, value attribute
- ApplicationContext, ClassPathXmlApplicationContext
- What exactly happens when context.getBean() is invoked

Code:

Country.java

```java
package com.cognizant.springlearn;

import org.slf4j.Logger;
import org.slf4j.LoggerFactory;

public class Country {
    private static final Logger LOGGER = LoggerFactory.getLogger(Country.class);

    private String code;
    private String name;

    public Country() {
        LOGGER.debug("Inside Country Constructor.");
    }

    public String getCode() {
        LOGGER.debug("Getter for code called");
        return code;
    }

    public void setCode(String code) {
        LOGGER.debug("Setter for code called");
        this.code = code;
    }

    public String getName() {
        LOGGER.debug("Getter for name called");
        return name;
```

```
    }

    public void setName(String name) {
        LOGGER.debug("Setter for name called");
        this.name = name;
    }

    @Override
    public String toString() {
        return "Country [code=" + code + ", name=" + name + "]";
    }
}
```

Country.xml

```
<?xml version="1.0" encoding="UTF-8"?>
<beans xmlns="http://www.springframework.org/schema/beans"
       xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
       xsi:schemaLocation="http://www.springframework.org/schema/beans
                           http://www.springframework.org/schema/beans/spring-
beans.xsd">

    <bean id="country" class="com.cognizant.springlearn.Country">
        <property name="code" value="IN" />
        <property name="name" value="India" />
    </bean>

</beans>
```

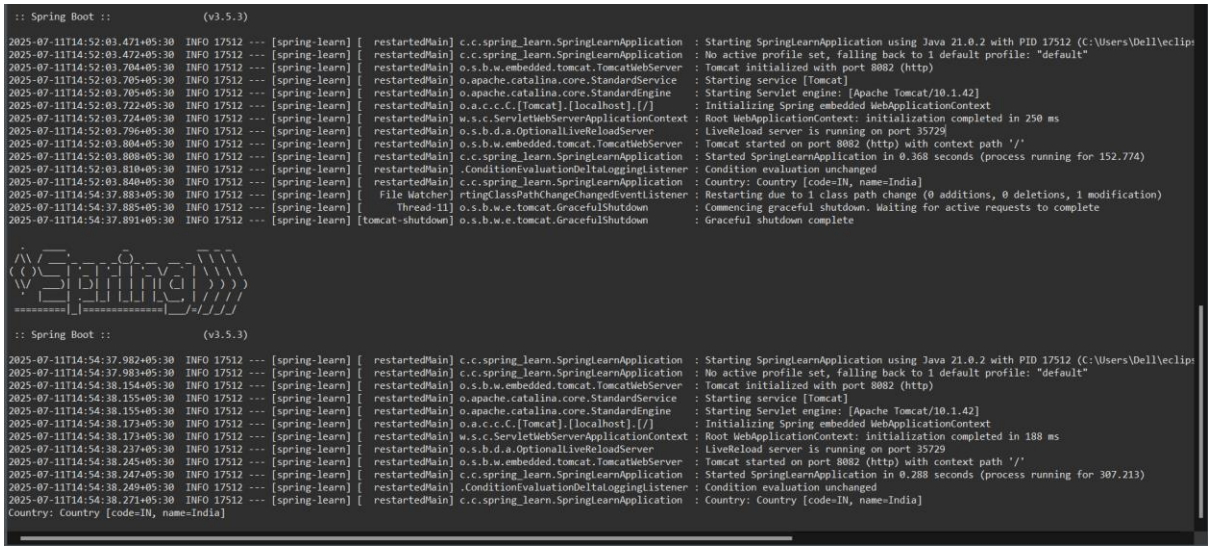
SpringLearnApplication.java

```
package com.cognizant.springlearn;

import org.springframework.boot.SpringApplication;
import org.springframework.boot.autoconfigure.SpringBootApplication;
import org.slf4j.Logger;
import org.slf4j.LoggerFactory;
import org.springframework.context.ApplicationContext;
import org.springframework.context.support.ClassPathXmlApplicationContext;
@SpringBootApplication
public class SpringLearnApplication {
    private static final Logger LOGGER =
LoggerFactory.getLogger(SpringLearnApplication.class);

    public static void main(String[] args) {
        SpringApplication.run(SpringLearnApplication.class, args);
        displayCountry();
    }
    public static void displayCountry() {
        ApplicationContext context = new
ClassPathXmlApplicationContext("country.xml");
        Country country = (Country) context.getBean("country", Country.class);
        LOGGER.debug("Country : {}", country.toString());
    }
}
```

Output:



```
 :: Spring Boot ::                (v3.5.3)

2025-07-11T14:52:03.471+05:30  INFO 17512 --- [spring-learn] [  restartedMain] c.c.spring_learn.SpringLearnApplication  : Starting SpringLearnApplication using Java 21.0.2 with PID 17512 (C:\Users\Dell\eclip
2025-07-11T14:52:03.472+05:30  INFO 17512 --- [spring-learn] [  restartedMain] c.c.spring_learn.SpringLearnApplication  : No active profile set, falling back to 1 default profile: "default"
2025-07-11T14:52:03.704+05:30  INFO 17512 --- [spring-learn] [  restartedMain] o.s.b.w.embedded.tomcat.TomcatWebServer  : Tomcat initialized with port 8082 (http)
2025-07-11T14:52:03.705+05:30  INFO 17512 --- [spring-learn] [  restartedMain] o.apache.catalina.core.StandardService   : Starting service [Tomcat]
2025-07-11T14:52:03.705+05:30  INFO 17512 --- [spring-learn] [  restartedMain] o.apache.catalina.core.StandardEngine    : Starting Servlet engine: [Apache Tomcat/10.1.42]
2025-07-11T14:52:03.722+05:30  INFO 17512 --- [spring-learn] [  restartedMain] o.a.c.c.C.[Tomcat].[localhost].[/]       : Initializing Spring embedded WebApplicationContext
2025-07-11T14:52:03.724+05:30  INFO 17512 --- [spring-learn] [  restartedMain] w.s.c.ServletWebServerApplicationContext : Root WebApplicationContext: initialization completed in 250 ms
2025-07-11T14:52:03.796+05:30  INFO 17512 --- [spring-learn] [  restartedMain] o.s.b.d.a.OptionalLiveReloadServer       : LiveReload server is running on port 35729
2025-07-11T14:52:03.804+05:30  INFO 17512 --- [spring-learn] [  restartedMain] o.s.b.w.embedded.tomcat.TomcatWebServer  : Tomcat started on port 8082 (http) with context path '/'
2025-07-11T14:52:03.808+05:30  INFO 17512 --- [spring-learn] [  restartedMain] c.c.spring_learn.SpringLearnApplication  : Started SpringLearnApplication in 0.368 seconds (process running for 152.774)
2025-07-11T14:52:03.810+05:30  INFO 17512 --- [spring-learn] [  restartedMain] .ConditionEvaluationDeltaLoggingListener : Condition evaluation unchanged
2025-07-11T14:52:03.840+05:30  INFO 17512 --- [spring-learn] [  restartedMain] c.c.spring_learn.SpringLearnApplication  : Country: Country [code=IN, name=India]
2025-07-11T14:54:37.883+05:30  INFO 17512 --- [spring-learn] [    File Watcher] rtingClassPathChangeChangedEventListener : Restarting due to 1 class path change (0 additions, 0 deletions, 1 modification)
2025-07-11T14:54:37.885+05:30  INFO 17512 --- [spring-learn] [      Thread-11] o.s.b.w.e.tomcat.GracefulShutdown        : Commencing graceful shutdown. Waiting for active requests to complete
2025-07-11T14:54:37.891+05:30  INFO 17512 --- [spring-learn] [tomcat-shutdown] o.s.b.w.e.tomcat.GracefulShutdown        : Graceful shutdown complete

  .   ____          _            __ _ _
 /\\ / ___'_ __ _ _(_)_ __  __ _ \ \ \ \
( ( )\___ | '_ | '_| | '_ \/ _` | \ \ \ \
 \\/  ___)| |_)| | | | | || (_| |  ) ) ) )
  '  |____| .__|_| |_|_| |_\__, | / / / /
 =========|_|==============|___/=/_/_/_/

 :: Spring Boot ::                (v3.5.3)

2025-07-11T14:54:37.982+05:30  INFO 17512 --- [spring-learn] [  restartedMain] c.c.spring_learn.SpringLearnApplication  : Starting SpringLearnApplication using Java 21.0.2 with PID 17512 (C:\Users\Dell\eclip
2025-07-11T14:54:37.983+05:30  INFO 17512 --- [spring-learn] [  restartedMain] c.c.spring_learn.SpringLearnApplication  : No active profile set, falling back to 1 default profile: "default"
2025-07-11T14:54:38.154+05:30  INFO 17512 --- [spring-learn] [  restartedMain] o.s.b.w.embedded.tomcat.TomcatWebServer  : Tomcat initialized with port 8082 (http)
2025-07-11T14:54:38.155+05:30  INFO 17512 --- [spring-learn] [  restartedMain] o.apache.catalina.core.StandardService   : Starting service [Tomcat]
2025-07-11T14:54:38.155+05:30  INFO 17512 --- [spring-learn] [  restartedMain] o.apache.catalina.core.StandardEngine    : Starting Servlet engine: [Apache Tomcat/10.1.42]
2025-07-11T14:54:38.173+05:30  INFO 17512 --- [spring-learn] [  restartedMain] o.a.c.c.C.[Tomcat].[localhost].[/]       : Initializing Spring embedded WebApplicationContext
2025-07-11T14:54:38.173+05:30  INFO 17512 --- [spring-learn] [  restartedMain] w.s.c.ServletWebServerApplicationContext : Root WebApplicationContext: initialization completed in 188 ms
2025-07-11T14:54:38.237+05:30  INFO 17512 --- [spring-learn] [  restartedMain] o.s.b.d.a.OptionalLiveReloadServer       : LiveReload server is running on port 35729
2025-07-11T14:54:38.245+05:30  INFO 17512 --- [spring-learn] [  restartedMain] o.s.b.w.embedded.tomcat.TomcatWebServer  : Tomcat started on port 8082 (http) with context path '/'
2025-07-11T14:54:38.247+05:30  INFO 17512 --- [spring-learn] [  restartedMain] c.c.spring_learn.SpringLearnApplication  : Started SpringLearnApplication in 0.288 seconds (process running for 307.213)
2025-07-11T14:54:38.249+05:30  INFO 17512 --- [spring-learn] [  restartedMain] .ConditionEvaluationDeltaLoggingListener : Condition evaluation unchanged
2025-07-11T14:54:38.271+05:30  INFO 17512 --- [spring-learn] [  restartedMain] c.c.spring_learn.SpringLearnApplication  : Country: Country [code=IN, name=India]
Country: Country [code=IN, name=India]
```