**Hands on 1**

**Create a Spring Web Project using Maven**

1. Opened https://start.spring.io/

2. Selected Project as Maven.

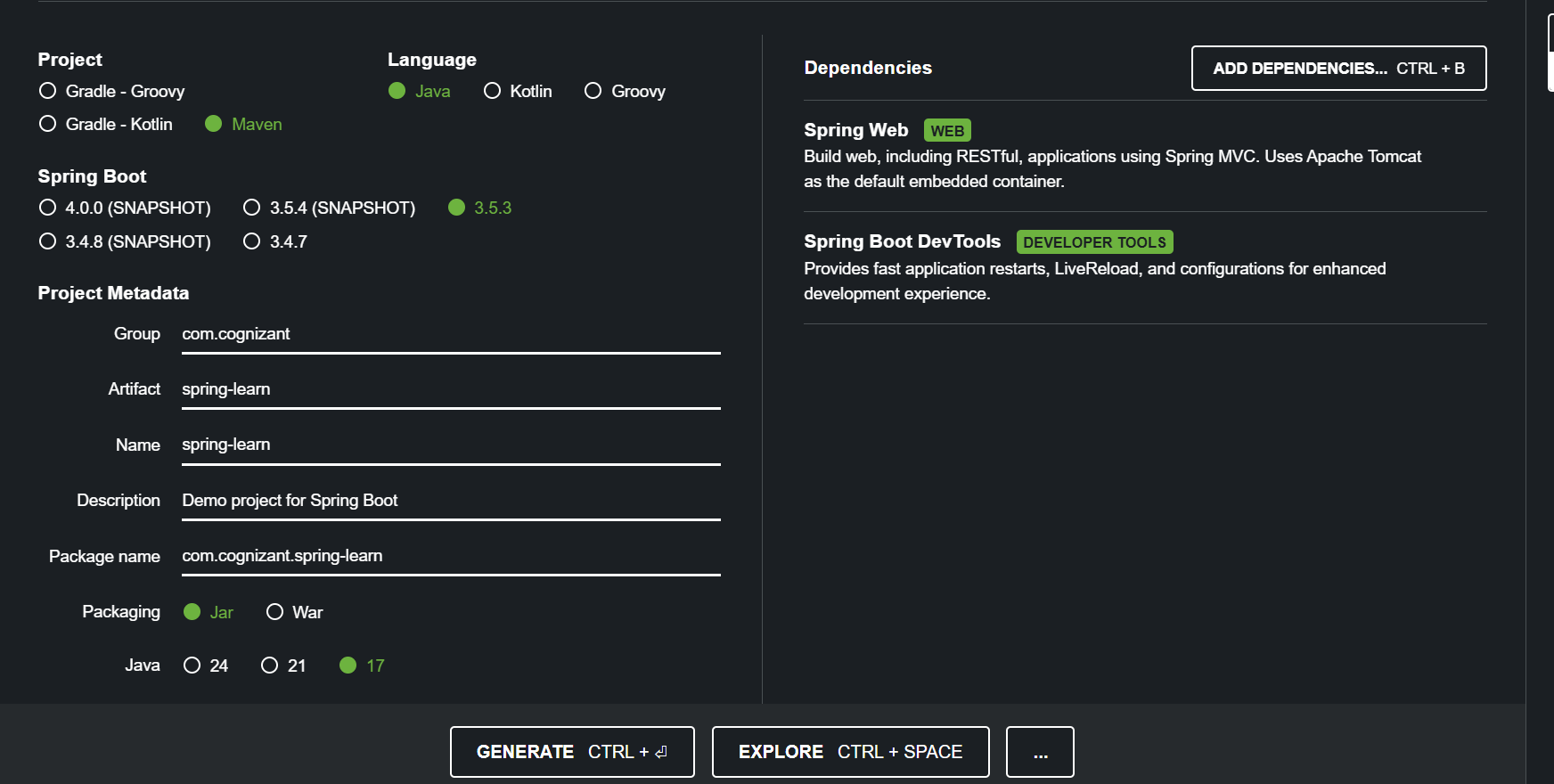
3. Set Group: com.cognizant

4. Set Artifact: spring-learn

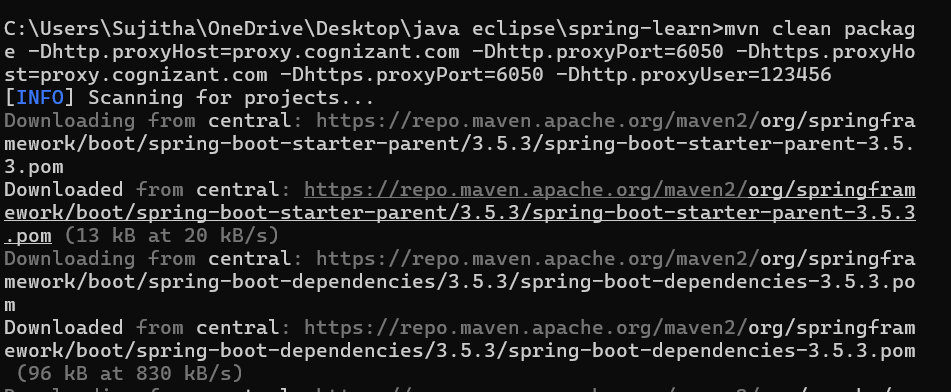
5. Added Dependencies: Spring Boot DevTools, Spring Web

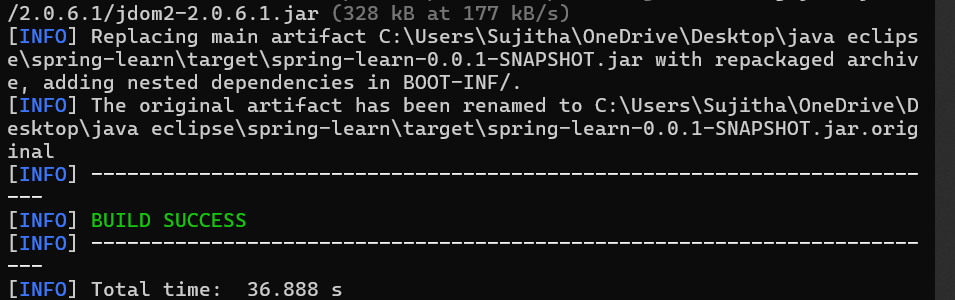
6. Clicked Generate and downloaded the ZIP.

7. Extracted the ZIP to Eclipse Workspace.



8.Built the project using the Maven command with proxy settings.





9.Imported the project in Eclipse using File > Import > Maven > Existing Maven Projects.

10. Verified the `main()` method in SpringLearnApplication.java by adding a log.

src/main/java/com/cognizant/SpringLearnApplication.java

package com.cognizant.spring\_learn;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

*@SpringBootApplication*

public class SpringLearnApplication {

public static void main(String[] args) {

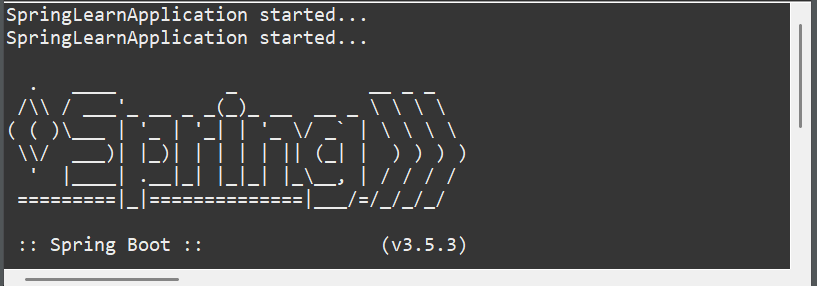
System.***out***.println("SpringLearnApplication started...");

SpringApplication.*run*(SpringLearnApplication.class, args);

}

}

Output



**Hands on 4**

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

src/main/java/com/cognizant/springlearn/Country.java

package com.cognizant.spring\_learn;

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("Inside getCode()");

return code;

}

public void setCode(String code) {

***LOGGER***.debug("Inside setCode()");

this.code = code;

}

public String getName() {

***LOGGER***.debug("Inside getName()");

return name;

}

public void setName(String name) {

***LOGGER***.debug("Inside setName()");

this.name = name;

}

*@Override*

public String toString() {

return "Country [code=" + code + ", name=" + name + "]";

}

}

src/main/resources/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.spring\_learn.Country"* >

<**property** name=*"code"* value=*"IN"* />

<**property** name=*"name"* value=*"India"* />

</**bean**>

</**beans**>

src/main/java/com/cognizant/springlearn/SpringLearnApplication.java

package com.cognizant.spring\_learn;

import org.slf4j.Logger;

import org.slf4j.LoggerFactory;

import org.springframework.context.ApplicationContext;

import org.springframework.context.support.ClassPathXmlApplicationContext;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

*@SpringBootApplication*

public class SpringLearnApplication {

private static final Logger ***LOGGER*** = LoggerFactory.*getLogger*(SpringLearnApplication.class);

public static void main(String[] args) {

***LOGGER***.debug("START");

SpringApplication.*run*(SpringLearnApplication.class, args);

*displayCountry*();

***LOGGER***.debug("END");

}

public static void displayCountry() {

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

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

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

}

}

Output

