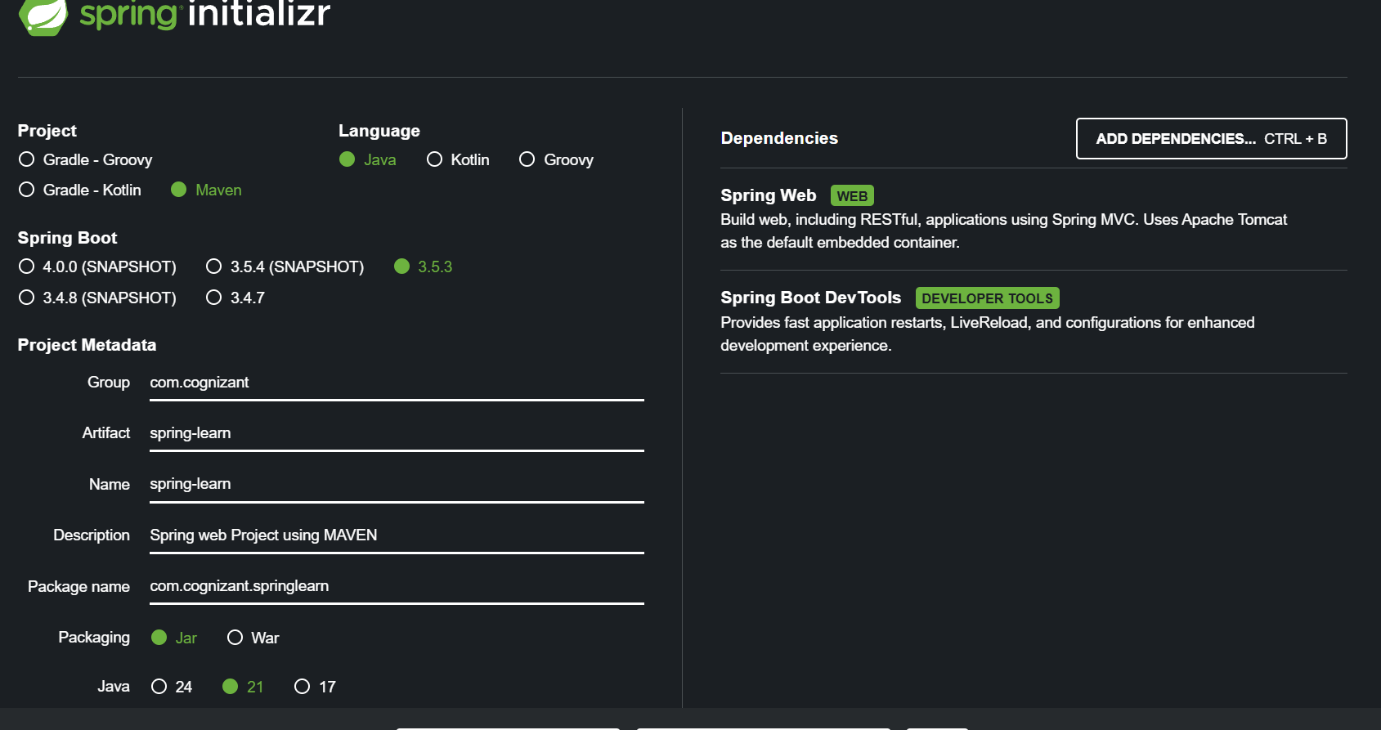
WEEK 4

Mandatory Hands-On

**Spring REST using Spring Boot**

**Create Spring Web project using MAVEN:**

****

* The project is downloaded as a .zip file and extracted in the Eclipse Workspace Directory.
* MAVEN is built in Command Prompt using the command

*mvn clean package -Dhttp.proxyHost=proxy.cognizant.com -Dhttp.proxyPort=6050 -Dhttps.proxyHost=proxy.cognizant.com -Dhttps.proxyPort=6050 -Dhttp.proxyUser=123456.*

* Then the Project is imported to Eclipse, run the project as Java Application.

**Main class:**

package com.cognizant.springlearn;

import org.springframework.boot.SpringApplication;

import org.springframework.boot.autoconfigure.SpringBootApplication;

*@SpringBootApplication*

public class SpringLearnApplication {

public static void main(String[] args) {

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

System.***out***.println("SpringLearnApplication using MAVEN has started...");

}

}

**pom.xml file:**

<?xml version="1.0" encoding="UTF-8"?>

<project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"

xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 https://maven.apache.org/xsd/maven-4.0.0.xsd">

<modelVersion>4.0.0</modelVersion>

<parent>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter-parent</artifactId>

<version>3.5.3</version>

<relativePath/> <!-- lookup parent from repository -->

</parent>

<groupId>com.cognizant</groupId>

<artifactId>spring-learn</artifactId>

<version>0.0.1-SNAPSHOT</version>

<name>spring-learn</name>

<description>Spring web Project using MAVEN</description>

<url/>

<licenses>

<license/>

</licenses>

<developers>

<developer/>

</developers>

<scm>

<connection/>

<developerConnection/>

<tag/>

<url/>

</scm>

<properties>

<java.version>21</java.version>

</properties>

<dependencies>

<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter-web</artifactId>

</dependency>

<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-devtools</artifactId>

<scope>runtime</scope>

<optional>true</optional>

</dependency>

<dependency>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-starter-test</artifactId>

<scope>test</scope>

</dependency>

</dependencies>

<build>

<plugins>

<plugin>

<groupId>org.springframework.boot</groupId>

<artifactId>spring-boot-maven-plugin</artifactId>

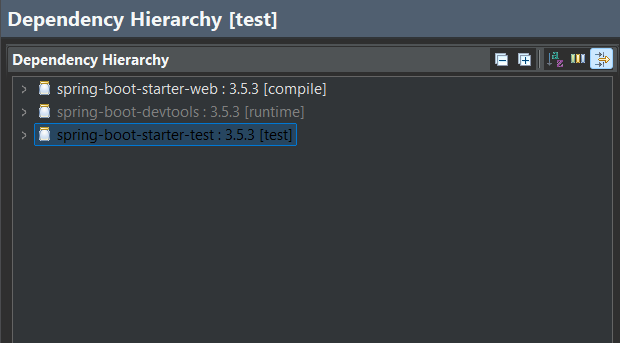
</plugin>

</plugins>

</build>

</project>

**SME Walkthrough:**

* src/ main/ java
  + Contains the main application code (e.g., controllers, services, models).
  + Entry point: SpringLearnApplication.java is located here.
* src/ main/ resources
  + application.properties or application.yml.
  + static folder → HTML/JS.
  + templates folder → Thymeleaf templates.
* src/ test/ java
  + Used for **unit** and **integration testing.**
  + Default test class: SpringLearnApplicationTests.java.
  + Use JUnit, Mockito, etc. for writing test cases.
* Purpose of @SpringBootApplication Annotation
  + @Configuration – Defines Spring Beans
  + @EnableAutoConfiguration – Auto-configures Spring application based on dependencies
  + @ComponentScan – Scans for components (@Component, @Service, @Repository, etc.)
* Dependency Hierarchy

**SAMPLE OUTPUT:**

