* 1. **Create a Spring Web Project using Maven**

**Objective:**

Setup a Spring Boot project using Maven in Spring Tool Suite and to prepare it for building REST APIs using spring web.

Steps in setting up Spring web project:

**Step1:** Generate Spring Boot Project using Spring Initializr

1. Firstly, go to <https://start.spring.io/> by opening in web browser.
2. Later, fill the project details as follows:

* Project: Maven
* Language: Java
* Spring Boot Version: 3.X
* Group: com.cognizant
* Artifact: spring-learn
* Dependencies:
* Spring Boot DevTools
* Spring Web

1. Click on the Generate. Then it generates a zip file with project name.
2. Extract the zip file int the sts workspace.

**Step2**: Import the project into Spring Tool Suite

1. Open Spring Tool Suite
2. Goto:

File > import > Existing Maven Projects

1. Browse and select the required folder.
2. Click finish to import the project.

**Step3:** Build the project using Maven

So, initially it generates a pom.xml. Include all the dependencies into the pom.xml

|  |
| --- |
| <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>3.2.5</version>  <relativePath/> </parent>  <groupId>com.cognizant</groupId>  <artifactId>spring-learn</artifactId>  <version>0.0.1-SNAPSHOT</version>  <name>spring-learn</name>  <description>Spring Boot Web Project</description>  <packaging>jar</packaging>  <properties>  <java.version>17</java.version>  </properties>  <dependencies>  <!-- Spring Boot Starter Web -->  <dependency>  <groupId>org.springframework. boot</groupId>  <artifactId>spring-boot-starter-web</artifactId>  </dependency>  <!-- Spring Boot DevTools (for hot reload) -->  <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>  <version>3.2.5</version>  </plugin>  </plugins>  </build>  </project> |

**Step4**: Locate the SpringLearnApplication.java file:

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

SpringLearnApplication.java:

|  |
| --- |
| package com.cognizant.spring\_learn;  import org.slf4j.Logger;  import org.slf4j.LoggerFactory;  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.info("START");          SpringApplication.run(SpringLearnApplication.class, args);          LOGGER.info("END");      }  } |

Also please locate HelloController.java in same path as well.

i.e.,

HelloController.java:

|  |
| --- |
| package com.cognizant.spring\_learn;  import org.springframework.web.bind.annotation.GetMapping;  import org.springframework.web.bind.annotation.RestController;  @RestController  public class HelloController {      @GetMapping("/hello")      public String sayHello() {          return "Hello from Spring Boot!";      }  } |

**Step5**: Now to display the outcome please create application.properties .

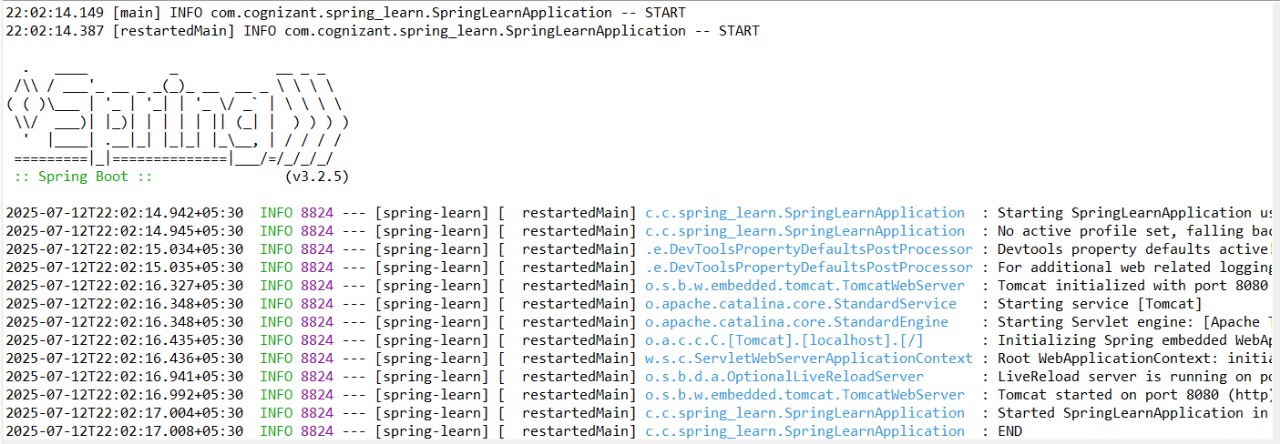
src/main/resources/application.properties

|  |
| --- |
| spring.application.name=spring-learn  server.port=8080 |

**Step6:** Run the Application

1. Now, Right click on SpringLearnApplication.java
2. Run as -> Spring Boot App.

**Expected Outcome**:



When we locate below URL in the console it displays as follows:

