

Spring Rest

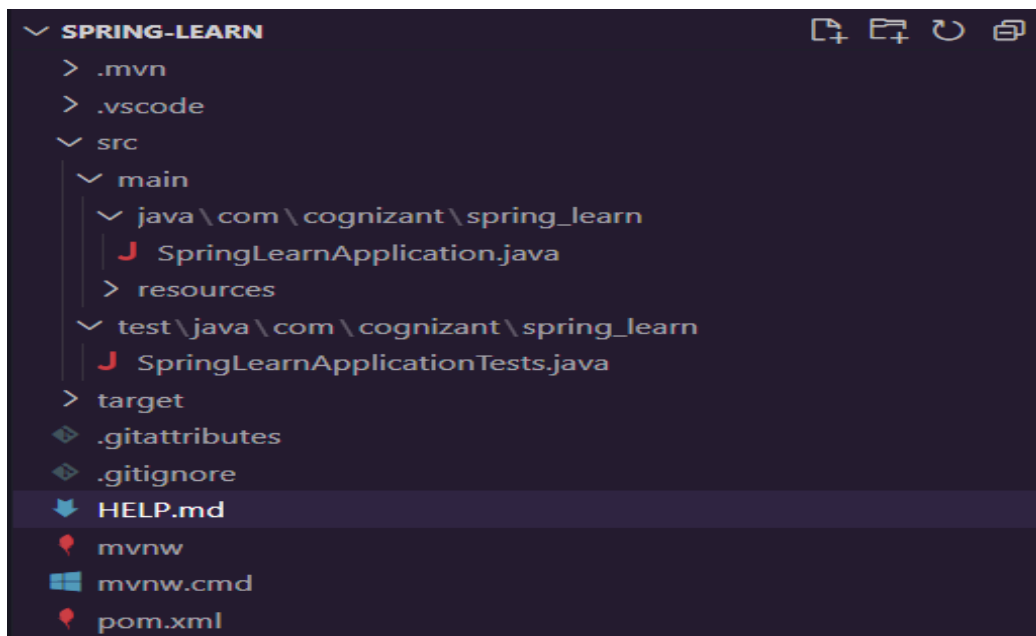
Create a Spring Web Project using Maven

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.

Program:

Structure:



Pom.xml:

```
<?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>Demo project for Spring Boot</description>
    <url/>
    <licenses>
        <license/>
    </licenses>
    <developers>
        <developer/>
    </developers>
    <scm>
        <connection/>
        <developerConnection/>
        <tag/>
        <url/>
    </scm>
    <properties>
        <java.version>21</java.version>
        <maven.compiler.source>21</maven.compiler.source>
        <maven.compiler.target>21</maven.compiler.target>
    </properties>
    <dependencies> Add Spring Boot Starters...
        <dependency>
            <groupId>org.springframework.boot</groupId>
            <artifactId>spring-boot-starter</artifactId>
        </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>
```

SpringLearnApplication.java:

```
package com.cognizant.spring_learn;

import org.springframework.boot.SpringApplication;
import org.springframework.boot.autoconfigure.SpringBootApplication;

@SpringBootApplication
public class SpringLearnApplication {

    Run | Debug
    public static void main(String[] args) {
        SpringApplication.run(primarySource:SpringLearnApplication.class, args);
    }
}
```

[illegible]

```

2023-07-09T08:20:42.374405:30 INFO 13488 -- [spring-launch] main c.c.s.SpringEarpApplicationTests -- Starting SpringEarpApplicationTests using Java 21.0.1 on PID 13488 (started by slvak in C:\Users\slvak\OneDrive\Desktop\Siva
ACTS-VIEWS-on-Spring-Code\Spring-Code)
2023-07-09T08:20:42.374405:30 INFO 13488 -- [spring-launch] [ main c.c.s.SpringEarpApplicationTests -- No active profile set, falling back to 1 default profile: 'default'
2023-07-09T08:20:42.484105:30 INFO 13488 -- [main c.c.s.SpringEarpApplicationTests -- Starting SpringEarpApplicationTests in 0.794 seconds (process running for 1.89)
INFO: Mockito is currently self-reflecting to enhance the initial mock-mocker. This will no longer work in future releases of the JDK. Please add Mockito as an agent to your build as described in Mockito's documentation: https://javadoc.io/doc/org.mockito/mockito-core-3.11.0-javadoc
WARNING: A Java agent has been loaded dynamically (C:\Users\slvak\AppData\Local\Temp\byuddy-byuddy-agent\1.17.6\byuddy-agent-1.17.6-6.jar)
WARNING: If a security tool is in use, please notify your vendor that JD-Ext-InstrumentClasspath is used for loading to hide this warning
WARNING: If a serviceability tool is not in use, please run with --Djdk-instrument-classpath for more information
WARNING: Dynamic loading of agents will be disabled by default in a future release
WARN: java.lang.NoClassDefFoundError: java/lang/ClassLoader$NativeImpl: warning: Sharing is only supported for boot loader classes because bootstrap classpath has been appended
[INFO] Tests run: 1, Failures: 0, Errors: 0, Skipped: 0, Time elapsed: 2.196 s -- in com.zigant.spring-launch.SpringEarpApplicationTests

```

```
[INFO] Tests run: 1, Failures: 0, Errors: 0, Skipped: 0
```

```
[INFO] --- spring-boot:3.5.3:repackage (repackage) @ spring-learn ---
[INFO] Replacing main artifact C:\Users\sivak\OneDrive\Desktop\Sivak\CTS\Hands-On\Spring\Core\spring-learn\target\spring-learn-0.0.1-SNAPSHOT.jar with repackaged archive, adding nested dependencies in BOOT-INF/.
[INFO] The original artifact has been renamed to C:\Users\sivak\OneDrive\Desktop\Sivak\CTS\Hands-On\Spring\Core\spring-learn\target\spring-learn-0.0.1-SNAPSHOT.jar.original
```

```
[INFO] BUILD SUCCESS
```

```
[INFO] -----
[INFO] Total time: 7.300 s
```

```
[INFO] Finished at: 2025-07-09T20:02:45+05:30
[INFO] -----
```

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.

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
                           https://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>
```

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(clazz:Country.class);

    private String code;
    private String name;

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

    public String getCode() {
        LOGGER.debug(msg:"Inside getCode()");
        return code;
    }

    public void setCode(String code) {
        LOGGER.debug(msg:"Inside setCode()");
        this.code = code;
    }

    public String getName() {
        LOGGER.debug(msg:"Inside getName()");
        return name;
    }

    public void setName(String name) {
        LOGGER.debug(msg:"Inside setName()");
        this.name = name;
    }

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

SpringLearnApplication.java

```
package com.cognizant.spring_learn;

import org.slf4j.Logger;
import org.slf4j.LoggerFactory;
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(clazz:SpringLearnApplication.class);

    Run | Debug
    public static void main(String[] args) {
        SpringApplication.run(primarySource:SpringLearnApplication.class, args);
        displayCountry();
    }

    public static void displayCountry() {
        LOGGER.info(msg:"START displayCountry()");
        try (ClassPathXmlApplicationContext context = new ClassPathXmlApplicationContext(configLocation:"country.xml")) {
            Country country = context.getBean(name:"country", requiredType:Country.class);
            LOGGER.debug(format:"Country : {}", country.toString());
        } catch (Exception e) {
            LOGGER.error(msg:"Error loading country bean: ", e);
        }
        LOGGER.info(msg:"END displayCountry()");
    }
}
```

Output:

```
  ____
 /  __ \
( (  __/
V  __/
'  __/
=====

:: Spring Boot ::      (v3.5.3)

2025-07-09T22:44:40.328+05:30 INFO 28260 --- [spring-learn] [main] c.c.spring_learn.SpringLearnApplication : Starting SpringLearnApplication using Java 21.0.1 with PID 28260 (C:\Users\sivak\OneDrive\Desktop\Siva\CTS\Hands-On\Spring Core\spring-learn\target\classes started by sivak in C:\Users\sivak\OneDrive\Desktop\Siva\CTS\Hands-On\Spring Core\spring-learn)
2025-07-09T22:44:40.331+05:30 INFO 28260 --- [spring-learn] [main] c.c.spring_learn.SpringLearnApplication : No active profile set, falling back to 1 default profile: "default"
2025-07-09T22:44:40.945+05:30 INFO 28260 --- [spring-learn] [main] c.c.spring_learn.SpringLearnApplication : Started SpringLearnApplication in 1.173 seconds (process running for 1.623)
2025-07-09T22:44:40.951+05:30 INFO 28260 --- [spring-learn] [main] c.c.spring_learn.SpringLearnApplication : START displayCountry()
2025-07-09T22:44:41.096+05:30 INFO 28260 --- [spring-learn] [main] c.c.spring_learn.SpringLearnApplication : Country : Country [code=IN, name=India]
2025-07-09T22:44:41.099+05:30 INFO 28260 --- [spring-learn] [main] c.c.spring_learn.SpringLearnApplication : END displayCountry()
```

Hello World RESTful Web Service

Write a REST service in the spring learn application created earlier, that returns the text "Hello World!!" using Spring Web Framework. Refer details below:

Method: GET

URL: /hello

Controller: com.cognizant.spring-learn.controller.HelloController

Method Signature: public String sayHello()

Method Implementation: return hard coded string "Hello World!!"

Sample Request: http://localhost:8083/hello

Sample Response: Hello World!!

IMPORTANT NOTE: Don't forget to include start and end log in the sayHello() method.

Try the URL http://localhost:8083/hello in both chrome browser and postman.

SME to explain the following aspects:

- In network tab of developer tools show the HTTP header details received
- In postman click on "Headers" tab to view the HTTP header details received

Program:

SpringLearnApplication.java:

```
package com.cognizant.spring_learn;

import org.springframework.boot.SpringApplication;
import org.springframework.boot.autoconfigure.SpringBootApplication;

@SpringBootApplication
public class SpringLearnApplication {

    Run | Debug
    public static void main(String[] args) {
        SpringApplication.run(primarySource:SpringLearnApplication.class, args);
    }
}
```

HelloController.java:

```
package com.cognizant.spring_learn.controller;

import org.slf4j.Logger;
import org.slf4j.LoggerFactory;
import org.springframework.web.bind.annotation.GetMapping;
import org.springframework.web.bind.annotation.RestController;

@RestController
public class HelloController {

    private static final Logger LOGGER = LoggerFactory.getLogger(clazz:HelloController.class);

    @GetMapping("/hello")
    public String sayHello() {
        LOGGER.info(msg:"START - sayHello()");
        String response = "Hello World!!";
        LOGGER.info(msg:"END - sayHello()");
        return response;
    }
}
```

pom.xml:

```
<dependency>
  <groupId>org.springframework.boot</groupId>
  <artifactId>spring-boot-starter-web</artifactId>
</dependency>
```

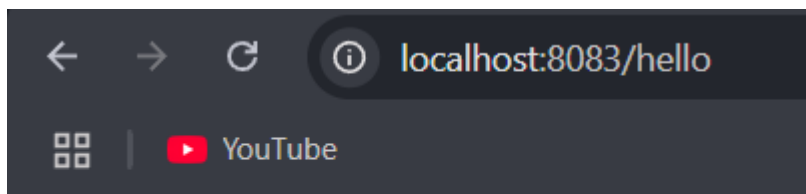
Output:

```

  ____  _
 / ___|| | | |
| |___| |_| |
 \___|_____|_|_|

:: Spring Boot ::      (v3.5.3)

2025-07-09T23:14:53.954+05:30 INFO 16280 --- [spring-learn] [    main] c.c.spring_learn.SpringLearnApplication : Starting SpringLearnApplication using Java 21.0.1 with PID 16280 (C:\Users\siva
k\OneDrive\Desktop\Siva\CTS\Hands-On\Spring Core\spring-learn\target\classes started by siva in C:\Users\siva\OneDrive\Desktop\Siva\CTS\Hands-On\Spring Core\spring-learn)
2025-07-09T23:14:53.959+05:30 INFO 16280 --- [spring-learn] [    main] c.c.spring_learn.SpringLearnApplication : No active profile set, falling back to 1 default profile: "default"
2025-07-09T23:14:55.316+05:30 INFO 16280 --- [spring-learn] [    main] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat initialized with port 8083 (http)
2025-07-09T23:14:55.331+05:30 INFO 16280 --- [spring-learn] [    main] o.apache.catalina.core.StandardService : Starting service [Tomcat]
2025-07-09T23:14:55.333+05:30 INFO 16280 --- [spring-learn] [    main] o.apache.catalina.core.StandardEngine : Starting Servlet engine: [Apache Tomcat/10.1.42]
2025-07-09T23:14:55.408+05:30 INFO 16280 --- [spring-learn] [    main] o.a.c.c.C.[Tomcat].[localhost].[/]   : Initializing Spring embedded WebApplicationContext
2025-07-09T23:14:55.411+05:30 INFO 16280 --- [spring-learn] [    main] w.s.c.ServletWebServerApplicationContext : Root WebApplicationContext: initialization completed in 1384 ms
2025-07-09T23:14:55.926+05:30 INFO 16280 --- [spring-learn] [    main] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat started on port 8083 (http) with context path '/'
2025-07-09T23:14:55.941+05:30 INFO 16280 --- [spring-learn] [    main] c.c.spring_learn.SpringLearnApplication : Started SpringLearnApplication in 2.692 seconds (process running for 3.149)
```



Hello World!!

REST- Country Web Service

Write a REST service that returns India country details in the earlier created spring learn application.

URL: /country

Controller: com.cognizant.spring-learn.controller.CountryController

Method Annotation: @RequestMapping

Method Name: getCountryIndia()

Method Implementation: Load India bean from spring xml configuration and return

Sample Request: http://localhost:8083/country

Sample Response:

```
{
  "code": "IN",
  "name": "India"
}
```

SME to explain the following aspects:

- What happens in the controller method?
- How the bean is converted into JSON response?
- In network tab of developer tools show the HTTP header details received
- In postman click on "Headers" tab to view the HTTP header details received

Program:

SpringLearnApplication.java:

```
package com.cognizant.spring_learn;

import org.springframework.boot.SpringApplication;
import org.springframework.boot.autoconfigure.SpringBootApplication;

@SpringBootApplication
public class SpringLearnApplication {

    Run | Debug
    public static void main(String[] args) {
        SpringApplication.run(primarySource:SpringLearnApplication.class, args);
    }
}
```

CountryController.java:

```
package com.cognizant.spring_learn.controller;

import com.cognizant.spring_learn.model.Country;
import org.slf4j.Logger;
import org.slf4j.LoggerFactory;
import org.springframework.context.ApplicationContext;
import org.springframework.context.support.ClassPathXmlApplicationContext;
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.bind.annotation.RestController;

@RestController
public class CountryController {

    private static final Logger LOGGER = LoggerFactory.getLogger(clazz:CountryController.class);

    @RequestMapping("/country")
    public Country getCountryIndia() {
        LOGGER.info(msg:"START - getCountryIndia()");

        ApplicationContext context = new ClassPathXmlApplicationContext(configLocation:"country.xml");
        Country country = (Country) context.getBean(name:"in");

        LOGGER.info(msg:"END - getCountryIndia()");
        return country;
    }
}
```

pom.xml:

```
<dependency>
  <groupId>org.springframework.boot</groupId>
  <artifactId>spring-boot-starter-web</artifactId>
</dependency>
```

Output:





```



  ____  _
 / ___|| | | |
| |___| |_| |
 \___ \|  __/
      |_|_|_|


:: Spring Boot ::      (v3.5.3)

- On [Spring Core] (spring-learn)
2025-07-09T23:47:22.004+05:30 INFO 15532 --- [spring-learn] [
2025-07-09T23:47:23.204+05:30 INFO 15532 --- [spring-learn] [
2025-07-09T23:47:23.227+05:30 INFO 15532 --- [spring-learn] [
2025-07-09T23:47:23.227+05:30 INFO 15532 --- [spring-learn] [
2025-07-09T23:47:23.318+05:30 INFO 15532 --- [spring-learn] [
2025-07-09T23:47:23.319+05:30 INFO 15532 --- [spring-learn] [
2025-07-09T23:47:23.846+05:30 INFO 15532 --- [spring-learn] [
2025-07-09T23:47:23.861+05:30 INFO 15532 --- [spring-learn] [

main] c.c.spring_learn.SpringLearnApplication : No active profile set, falling back to 1 default profile: "default"
main] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat initialized with port 8083 (http)
main] o.apache.catalina.core.StandardService : Starting service [Tomcat]
main] o.apache.catalina.core.StandardEngine : Starting Servlet engine: [Apache Tomcat/10.1.42]
main] o.a.c.c.C.[Tomcat].[localhost].[/] : Initializing Spring embedded WebApplicationContext
main] w.s.c.ServletWebServerApplicationContext : Root WebApplicationContext: initialization completed in 1251 ms
main] o.s.b.w.embedded.tomcat.TomcatWebServer : Tomcat started on port 8083 (http) with context path '/'
main] c.c.spring_learn.SpringLearnApplication : Started SpringLearnApplication in 2.393 seconds (process running for 2.817)
```

 localhost:8083/country

 |  YouTube

Pretty-print 

```
{
  "code": "IN",
  "name": "India"
}
```

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.

Program:

HomeController.java

```
package com.example.controller;

import org.springframework.stereotype.Controller;
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.ui.Model;

@Controller
public class HomeController {

    @RequestMapping("/")
    public String home(Model model) {
        model.addAttribute("message", "Welcome to Spring MVC!");
        return "home";
    }
}
```

web.xml:

```
package com.example.controller;

import org.springframework.stereotype.Controller;
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.ui.Model;

@Controller
public class HomeController {

    @RequestMapping("/")
    public String home(Model model) {
        model.addAttribute("message", "Welcome to Spring MVC!");
        return "home";
    }
}
```

dispatcher-servlet.xml

```
<beans xmlns="http://www.springframework.org/schema/beans"
        xmlns:mvc="http://www.springframework.org/schema/mvc"
        xmlns:context="http://www.springframework.org/schema/context"
        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
            http://www.springframework.org/schema/context
            http://www.springframework.org/schema/context/spring-context.xsd
            http://www.springframework.org/schema/mvc
            http://www.springframework.org/schema/mvc/spring-mvc.xsd">

    <context:component-scan base-package="com.example.controller" />
    <mvc:annotation-driven />

    <bean
class="org.springframework.web.servlet.view.InternalResourceViewResolver">
        <property name="prefix" value="/WEB-INF/views/" />
        <property name="suffix" value=".jsp" />
    </bean>

</beans>
```

Home.jsp:

```
%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
<html>
  <head><title>Home Page</title></head>
  <body>
    <h2>${message}</h2>
  </body>
</html>
```

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
    http://maven.apache.org/xsd/maven-4.0.0.xsd">

  <modelVersion>4.0.0</modelVersion>
  <groupId>com.example</groupId>
  <artifactId>spring-web-demo</artifactId>
  <version>1.0-SNAPSHOT</version>
  <packaging>war</packaging>

  <name>Spring Web Demo</name>

  <dependencies>
    <!-- Spring Web MVC -->
    <dependency>
      <groupId>org.springframework</groupId>
      <artifactId>spring-webmvc</artifactId>
      <version>5.3.32</version>
    </dependency>

    <!-- JSTL -->
    <dependency>
      <groupId>javax.servlet</groupId>
      <artifactId>jstl</artifactId>
      <version>1.2</version>
    </dependency>

    <!-- Servlet API -->
    <dependency>
      <groupId>javax.servlet</groupId>
      <artifactId>javax.servlet-api</artifactId>
      <version>4.0.1</version>
      <scope>provided</scope>
    </dependency>
  </dependencies>

  <build>
    <finalName>spring-web-demo</finalName>
  </build>
</project>
```


Output:

```
[INFO] Scanning for projects...
[INFO]
[INFO] -----< com.example:spring-web-demo >-----
[INFO] Building Spring Web Demo 1.0-SNAPSHOT
[INFO] from pom.xml
[INFO] -----[ war ]-----
Downloading from central: https://repo.maven.apache.org/maven2/org/apache/maven/plugins/maven-war-plugin/3.4.0/maven-war-plugin-3.4.0.pom
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/plugins/maven-war-plugin/3.4.0/maven-war-plugin-3.4.0.pom (8.4 kB at 7.7 kB/s)
Downloading from central: https://repo.maven.apache.org/maven2/org/apache/maven/plugins/maven-war-plugin/3.4.0/maven-war-plugin-3.4.0.jar
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/plugins/maven-war-plugin/3.4.0/maven-war-plugin-3.4.0.jar (83 kB at 452 kB/s)
Downloading from central: https://repo.maven.apache.org/maven2/org/springframework/spring-webmvc/5.3.32/spring-webmvc-5.3.32.pom
Downloaded from central: https://repo.maven.apache.org/maven2/org/springframework/spring-webmvc/5.3.32/spring-webmvc-5.3.32.pom (3.0 kB at 63 kB/s)
Downloading from central: https://repo.maven.apache.org/maven2/org/springframework/spring-aop/5.3.32/spring-aop-5.3.32.pom
Downloaded from central: https://repo.maven.apache.org/maven2/org/springframework/spring-aop/5.3.32/spring-aop-5.3.32.pom (2.2 kB at 39 kB/s)
Downloading from central: https://repo.maven.apache.org/maven2/org/springframework/spring-beans/5.3.32/spring-beans-5.3.32.pom
Downloaded from central: https://repo.maven.apache.org/maven2/org/springframework/spring-beans/5.3.32/spring-beans-5.3.32.pom (2.0 kB at 33 kB/s)
Downloading from central: https://repo.maven.apache.org/maven2/org/springframework/spring-core/5.3.32/spring-core-5.3.32.pom
Downloaded from central: https://repo.maven.apache.org/maven2/org/springframework/spring-core/5.3.32/spring-core-5.3.32.pom (2.0 kB at 40 kB/s)
Downloading from central: https://repo.maven.apache.org/maven2/org/springframework/spring-jcl/5.3.32/spring-jcl-5.3.32.pom
Downloaded from central: https://repo.maven.apache.org/maven2/org/springframework/spring-jcl/5.3.32/spring-jcl-5.3.32.pom (1.8 kB at 37 kB/s)
Downloading from central: https://repo.maven.apache.org/maven2/org/springframework/spring-context/5.3.32/spring-context-5.3.32.pom
Downloaded from central: https://repo.maven.apache.org/maven2/org/springframework/spring-context/5.3.32/spring-context-5.3.32.pom (2.6 kB at 40 kB/s)
Downloading from central: https://repo.maven.apache.org/maven2/org/springframework/spring-expression/5.3.32/spring-expression-5.3.32.pom
Downloaded from central: https://repo.maven.apache.org/maven2/org/springframework/spring-expression/5.3.32/spring-expression-5.3.32.pom (2.1 kB at 32 kB/s)
Downloading from central: https://repo.maven.apache.org/maven2/org/springframework/spring-web/5.3.32/spring-web-5.3.32.pom
Downloaded from central: https://repo.maven.apache.org/maven2/org/springframework/spring-web/5.3.32/spring-web-5.3.32.pom (2.2 kB at 37 kB/s)
Downloading from central: https://repo.maven.apache.org/maven2/org/javax/servlet/jstl/1.2/jstl-1.2.pom
Downloaded from central: https://repo.maven.apache.org/maven2/org/javax/servlet/jstl/1.2/jstl-1.2.pom (356 B at 7.7 kB/s)
Downloading from central: https://repo.maven.apache.org/maven2/org/javax/servlet/javax.servlet-api/4.0.1/javax.servlet-api-4.0.1.pom
Downloaded from central: https://repo.maven.apache.org/maven2/org/javax/servlet/javax.servlet-api/4.0.1/javax.servlet-api-4.0.1.pom (16 kB at 277 kB/s)
Downloading from central: https://repo.maven.apache.org/maven2/org/springframework/spring-webmvc/5.3.32/spring-webmvc-5.3.32.jar
Downloaded from central: https://repo.maven.apache.org/maven2/org/springframework/spring-webmvc/5.3.32/spring-webmvc-5.3.32.jar (1.0 MB at 2.4 MB/s)
Downloading from central: https://repo.maven.apache.org/maven2/org/springframework/spring-aop/5.3.32/spring-aop-5.3.32.jar
Downloaded from central: https://repo.maven.apache.org/maven2/org/springframework/spring-aop/5.3.32/spring-aop-5.3.32.jar
Downloading from central: https://repo.maven.apache.org/maven2/org/springframework/spring-beans/5.3.32/spring-beans-5.3.32.jar
Downloaded from central: https://repo.maven.apache.org/maven2/org/springframework/spring-beans/5.3.32/spring-beans-5.3.32.jar
Downloading from central: https://repo.maven.apache.org/maven2/org/springframework/spring-core/5.3.32/spring-core-5.3.32.jar
Downloaded from central: https://repo.maven.apache.org/maven2/org/springframework/spring-core/5.3.32/spring-core-5.3.32.jar
Downloading from central: https://repo.maven.apache.org/maven2/org/springframework/spring-jcl/5.3.32/spring-jcl-5.3.32.jar
Downloaded from central: https://repo.maven.apache.org/maven2/org/springframework/spring-jcl/5.3.32/spring-jcl-5.3.32.jar
Downloading from central: https://repo.maven.apache.org/maven2/org/springframework/spring-context/5.3.32/spring-context-5.3.32.jar
Downloaded from central: https://repo.maven.apache.org/maven2/org/springframework/spring-context/5.3.32/spring-context-5.3.32.jar
Downloading from central: https://repo.maven.apache.org/maven2/org/springframework/spring-expression/5.3.32/spring-expression-5.3.32.jar
Downloaded from central: https://repo.maven.apache.org/maven2/org/springframework/spring-expression/5.3.32/spring-expression-5.3.32.jar
Downloading from central: https://repo.maven.apache.org/maven2/org/springframework/spring-beans/5.3.32/spring-beans-5.3.32.jar
Downloaded from central: https://repo.maven.apache.org/maven2/org/springframework/spring-beans/5.3.32/spring-beans-5.3.32.jar (706 kB at 1.9 MB/s)
Downloading from central: https://repo.maven.apache.org/maven2/org/springframework/spring-web/5.3.32/spring-web-5.3.32.jar
Downloaded from central: https://repo.maven.apache.org/maven2/org/springframework/spring-web/5.3.32/spring-web-5.3.32.jar (385 kB at 603 kB/s)
Downloading from central: https://repo.maven.apache.org/maven2/org/javax/servlet/jstl/1.2/jstl-1.2.jar
Downloaded from central: https://repo.maven.apache.org/maven2/org/javax/servlet/jstl/1.2/jstl-1.2.jar (414 kB at 304 kB/s)
Downloading from central: https://repo.maven.apache.org/maven2/org/javax/servlet/javax.servlet-api/4.0.1/javax.servlet-api-4.0.1.jar
Downloaded from central: https://repo.maven.apache.org/maven2/org/javax/servlet/javax.servlet-api/4.0.1/javax.servlet-api-4.0.1.jar (96 kB at 69 kB/s)
Downloading from central: https://repo.maven.apache.org/maven2/org/springframework/spring-web/5.3.32/spring-web-5.3.32.jar
Downloaded from central: https://repo.maven.apache.org/maven2/org/springframework/spring-web/5.3.32/spring-web-5.3.32.jar (1.6 MB at 1.2 MB/s)
Downloading from central: https://repo.maven.apache.org/maven2/org/springframework/spring-core/5.3.32/spring-core-5.3.32.jar
Downloaded from central: https://repo.maven.apache.org/maven2/org/springframework/spring-core/5.3.32/spring-core-5.3.32.jar (1.5 MB at 810 kB/s)
Downloading from central: https://repo.maven.apache.org/maven2/org/springframework/spring-context/5.3.32/spring-context-5.3.32.jar
Downloaded from central: https://repo.maven.apache.org/maven2/org/springframework/spring-context/5.3.32/spring-context-5.3.32.jar (1.3 MB at 677 kB/s)
[INFO]
[INFO] --- clean:3.2.0:clean (default-clean) @ spring-web-demo ---
[INFO]
[INFO] --- resources:3.3.1:resources (default-resources) @ spring-web-demo ---
[WARNING] Using platform encoding (UTF-8 actually) to copy filtered resources, i.e. build is platform dependent!
[INFO] Copying 0 resource from src/main/resources to target/classes
[INFO]
[INFO] --- compiler:3.13.0:compile (default-compile) @ spring-web-demo ---
[INFO] Recompiling the module because of changed dependency.
[WARNING] File encoding has not been set, using platform encoding UTF-8, i.e. build is platform dependent!
[INFO] Compiling 1 source file with javac [debug target 1.8] to target/classes
[WARNING] bootstrap class path not set in conjunction with -source 8
[WARNING] source value 8 is obsolete and will be removed in a future release
[WARNING] target value 8 is obsolete and will be removed in a future release
[WARNING] To suppress warnings about obsolete options, use -Xlint:-options.
[INFO]
[INFO] --- resources:3.3.1:testResources (default-testResources) @ spring-web-demo ---
[WARNING] Using platform encoding (UTF-8 actually) to copy filtered resources, i.e. build is platform dependent!
[INFO] skip non existing resourceDirectory E:\CTS\WEEK_04\spring-web-demo\src\test\resources
[INFO]
[INFO] --- compiler:3.13.0:testCompile (default-testCompile) @ spring-web-demo ---
[INFO] No sources to compile
[INFO]
[INFO] --- surefire:3.2.5:test (default-test) @ spring-web-demo ---
[INFO] No tests to run.
[INFO]
[INFO] --- war:3.4.0:war (default-war) @ spring-web-demo ---
Downloading from central: https://repo.maven.apache.org/maven2/org/apache/maven/maven-archiver/3.6.0/maven-archiver-3.6.0.pom
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/maven-archiver/3.6.0/maven-archiver-3.6.0.pom (3.9 kB at 98 kB/s)
Downloading from central: https://repo.maven.apache.org/maven2/org/codehaus/plexus/plexus-io/3.4.0/plexus-io-3.4.0.pom
Downloaded from central: https://repo.maven.apache.org/maven2/org/codehaus/plexus/plexus-io/3.4.0/plexus-io-3.4.0.pom (6.0 kB at 150 kB/s)
Downloading from central: https://repo.maven.apache.org/maven2/org/codehaus/plexus/plexus-archiver/4.4.0/plexus-archiver-4.4.0.pom
Downloaded from central: https://repo.maven.apache.org/maven2/org/codehaus/plexus/plexus-archiver/4.4.0/plexus-archiver-4.4.0.pom (6.3 kB at 72 kB/s)
```

```
Downloaded from central: https://repo.maven.apache.org/maven2/org/springframework/spring-jcl/5.3.32/spring-jcl-5.3.32.jar (25 kB at 120 kB/s)
Downloading from central: https://repo.maven.apache.org/maven2/org/springframework/spring-expression/5.3.32/spring-expression-5.3.32.jar
Downloaded from central: https://repo.maven.apache.org/maven2/org/springframework/spring-expression/5.3.32/spring-expression-5.3.32.jar (25 kB at 120 kB/s)
Downloading from central: https://repo.maven.apache.org/maven2/org/springframework/spring-beans/5.3.32/spring-beans-5.3.32.jar (706 kB at 1.9 MB/s)
Downloaded from central: https://repo.maven.apache.org/maven2/org/springframework/spring-beans/5.3.32/spring-beans-5.3.32.jar (706 kB at 1.9 MB/s)
Downloading from central: https://repo.maven.apache.org/maven2/org/springframework/spring-web/5.3.32/spring-web-5.3.32.jar
Downloaded from central: https://repo.maven.apache.org/maven2/org/springframework/spring-web/5.3.32/spring-web-5.3.32.jar (385 kB at 603 kB/s)
Downloading from central: https://repo.maven.apache.org/maven2/org/javax/servlet/jstl/1.2/jstl-1.2.jar
Downloaded from central: https://repo.maven.apache.org/maven2/org/javax/servlet/jstl/1.2/jstl-1.2.jar (414 kB at 304 kB/s)
Downloading from central: https://repo.maven.apache.org/maven2/org/javax/servlet/javax.servlet-api/4.0.1/javax.servlet-api-4.0.1.jar
Downloaded from central: https://repo.maven.apache.org/maven2/org/javax/servlet/javax.servlet-api/4.0.1/javax.servlet-api-4.0.1.jar (96 kB at 69 kB/s)
Downloading from central: https://repo.maven.apache.org/maven2/org/springframework/spring-web/5.3.32/spring-web-5.3.32.jar
Downloaded from central: https://repo.maven.apache.org/maven2/org/springframework/spring-web/5.3.32/spring-web-5.3.32.jar (1.6 MB at 1.2 MB/s)
Downloading from central: https://repo.maven.apache.org/maven2/org/springframework/spring-core/5.3.32/spring-core-5.3.32.jar
Downloaded from central: https://repo.maven.apache.org/maven2/org/springframework/spring-core/5.3.32/spring-core-5.3.32.jar (1.5 MB at 810 kB/s)
Downloading from central: https://repo.maven.apache.org/maven2/org/springframework/spring-context/5.3.32/spring-context-5.3.32.jar
Downloaded from central: https://repo.maven.apache.org/maven2/org/springframework/spring-context/5.3.32/spring-context-5.3.32.jar (1.3 MB at 677 kB/s)
```

```
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/commons/commons-parent/56/commons-parent-56.jar
Downloading from central: https://repo.maven.apache.org/maven2/org/junit/junit-bom/5.9.1/junit-bom-5.9.1.jar
Downloaded from central: https://repo.maven.apache.org/maven2/org/junit/junit-bom/5.9.1/junit-bom-5.9.1.jar
Downloading from central: https://repo.maven.apache.org/maven2/com/github/luben/zstd-jni/1.5.5-2/zstd-jni-1.5.5-2.jar
Downloaded from central: https://repo.maven.apache.org/maven2/com/github/luben/zstd-jni/1.5.5-2/zstd-jni-1.5.5-2.jar
Downloading from central: https://repo.maven.apache.org/maven2/org/apache/maven/shared/maven-mapping/3.3.0/maven-mapping-3.3.0.jar
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/shared/maven-mapping/3.3.0/maven-mapping-3.3.0.jar
Downloading from central: https://repo.maven.apache.org/maven2/org/apache/maven/maven-archiver/3.6.0/maven-archiver-3.6.0.jar
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/maven/maven-archiver/3.6.0/maven-archiver-3.6.0.jar
Downloading from central: https://repo.maven.apache.org/maven2/org/codehaus/plexus/plexus-io/3.4.0/plexus-io-3.4.0.jar
Downloaded from central: https://repo.maven.apache.org/maven2/org/codehaus/plexus/plexus-io/3.4.0/plexus-io-3.4.0.jar
Downloading from central: https://repo.maven.apache.org/maven2/org/codehaus/plexus/plexus-archiver/4.7.0/plexus-archiver-4.7.0.jar
Downloaded from central: https://repo.maven.apache.org/maven2/org/codehaus/plexus/plexus-archiver/4.7.0/plexus-archiver-4.7.0.jar
Downloading from central: https://repo.maven.apache.org/maven2/org/apache/commons/commons-compress/1.23/commons-compress-1.23.jar
Downloaded from central: https://repo.maven.apache.org/maven2/org/apache/commons/commons-compress/1.23/commons-compress-1.23.jar
Downloading from central: https://repo.maven.apache.org/maven2/com/github/luben/zstd-jni/1.5.5-2/zstd-jni-1.5.5-2.jar
Downloaded from central: https://repo.maven.apache.org/maven2/com/github/luben/zstd-jni/1.5.5-2/zstd-jni-1.5.5-2.jar
[INFO] Packaging webapp
[INFO] Assembling webapp [spring-web-demo] in [E:\CTS\WEEK_04\spring-web-demo\target\spring-web-demo]
[INFO] Processing war project
[INFO] Copying webapp resources [E:\CTS\WEEK_04\spring-web-demo\src\main\webapp]
[INFO] Building war: E:\CTS\WEEK_04\spring-web-demo\target\spring-web-demo.war
[INFO]
[INFO] --- install:3.1.2:install (default-install) @ spring-web-demo ---
[INFO] Installing E:\CTS\WEEK_04\spring-web-demo\pom.xml to C:\Users\Abhinithaa\.m2\repository\com\exam\spring-web-demo-1.0-SNAPSHOT.war
[INFO] Installing E:\CTS\WEEK_04\spring-web-demo\target\spring-web-demo.war to C:\Users\Abhinithaa\.m2\repository\com\exam\spring-web-demo-1.0-SNAPSHOT.war
[INFO] -----
[INFO] BUILD SUCCESS
[INFO] -----
[INFO] Total time: 22.261 s
[INFO] Finished at: 2025-07-08T19:11:56+05:30
[INFO] -----
```