Sprint Boot Explanation and Configuration for Basic Web Project

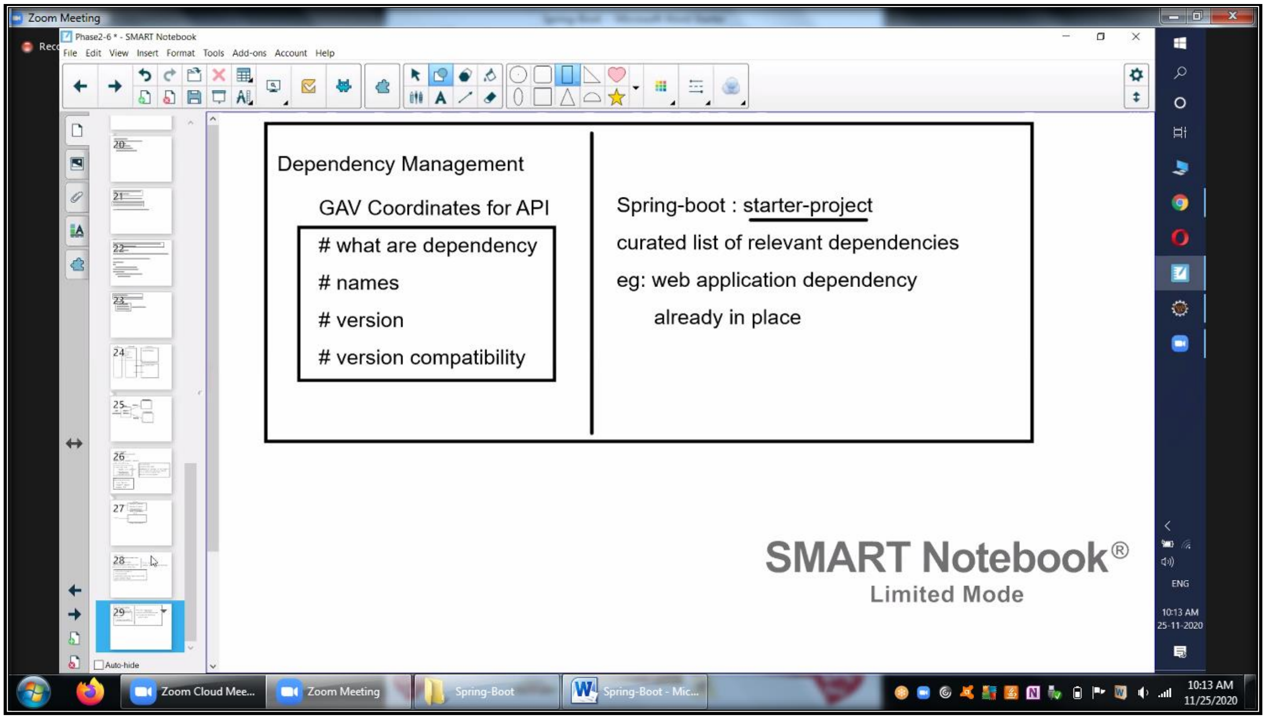
Existing Maven approach and Spring Boot approach for GAV coordinates

Just 1 GAV co-ordinate will take care of the entire dependencies

For Web – only 1 GAV co-ordinate

For Spring Security – only 1 GAV co-ordinate

For DB Sprint Security – only 1 GAV co-ordinate etc.



Dependency Management

GAV Coordinates for API

- # what are dependency
- # names
- # version
- # version compatibility

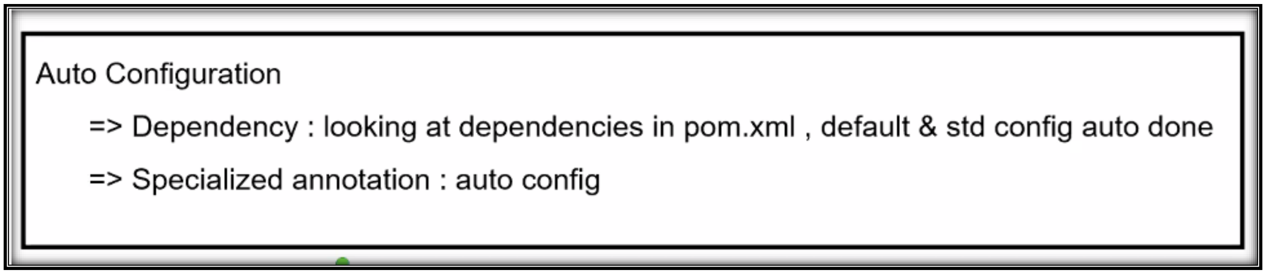
Spring-boot : starter-project

curated list of relevant dependencies

eg: web application dependency

already in place

Auto Configuration

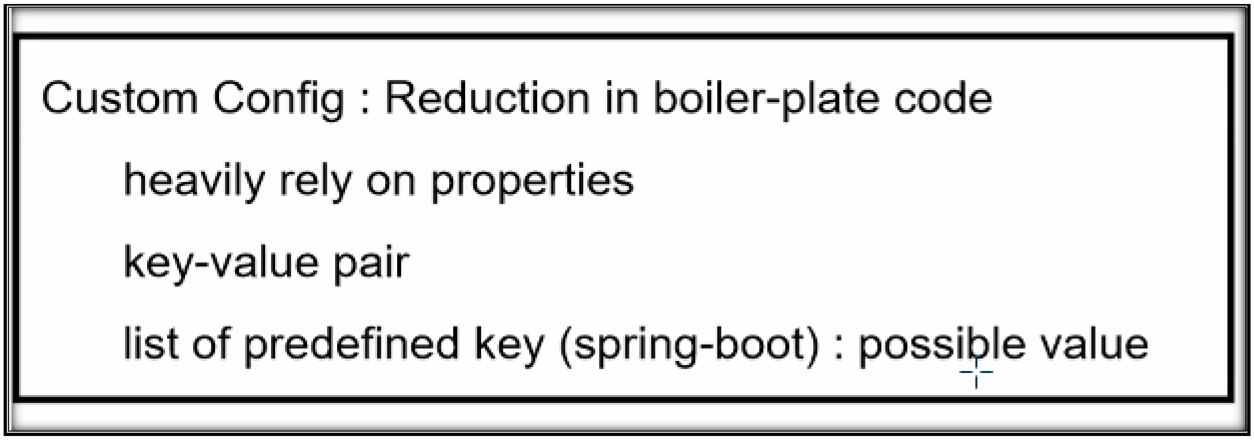


Auto Configuration

=> Dependency : looking at dependencies in pom.xml , default & std config auto done

=> Specialized annotation : auto config

Properties file is a key feature for Spring-Boot



Custom Config : Reduction in boiler-plate code

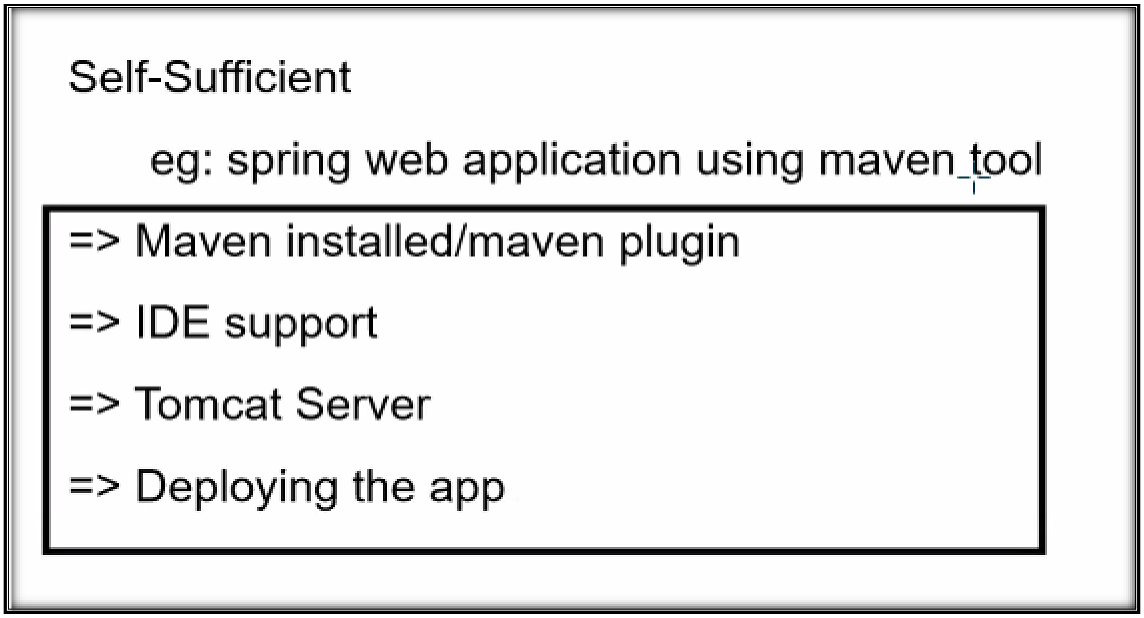
heavily rely on properties

key-value pair

list of predefined key (spring-boot) : possible value

Self-Sufficient

Current Approach



Self-Sufficient

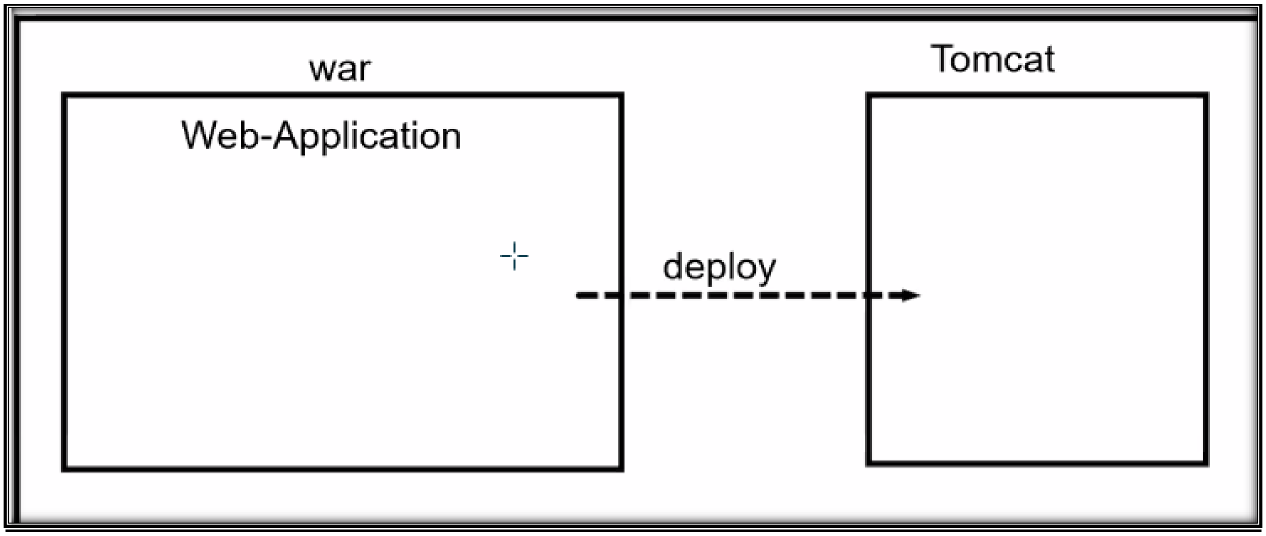
eg: spring web application using maven tool

=> Maven installed/maven plugin

=> IDE support

=> Tomcat Server

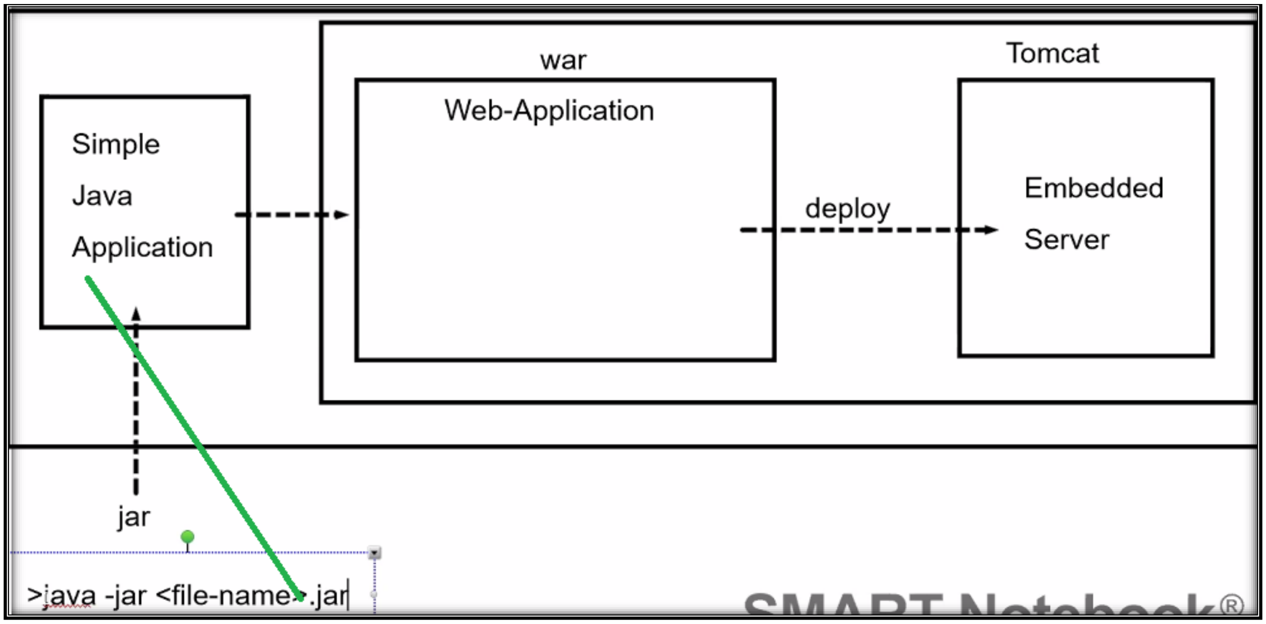
=> Deploying the app



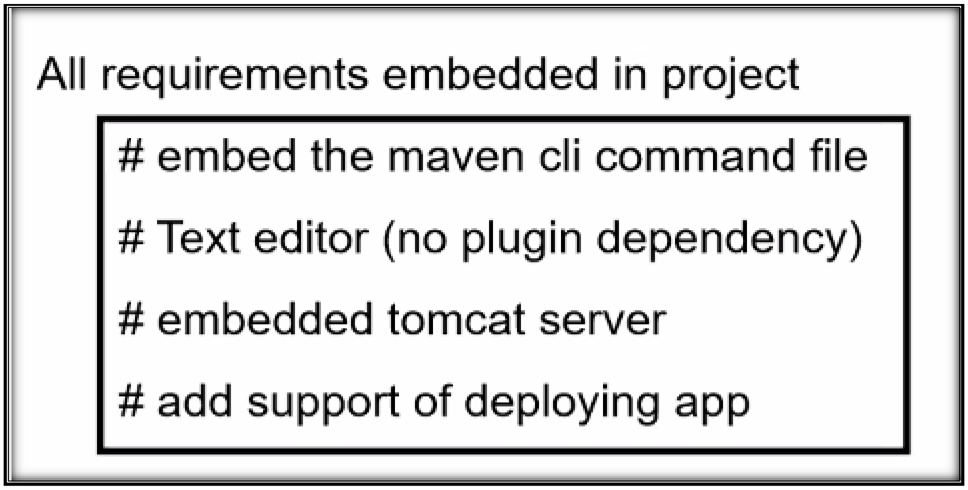
Spring-Boot Approach

Independent and help us to get all the things to get easily, Most of them are in-built

Simple Java file will take care of all the aspects

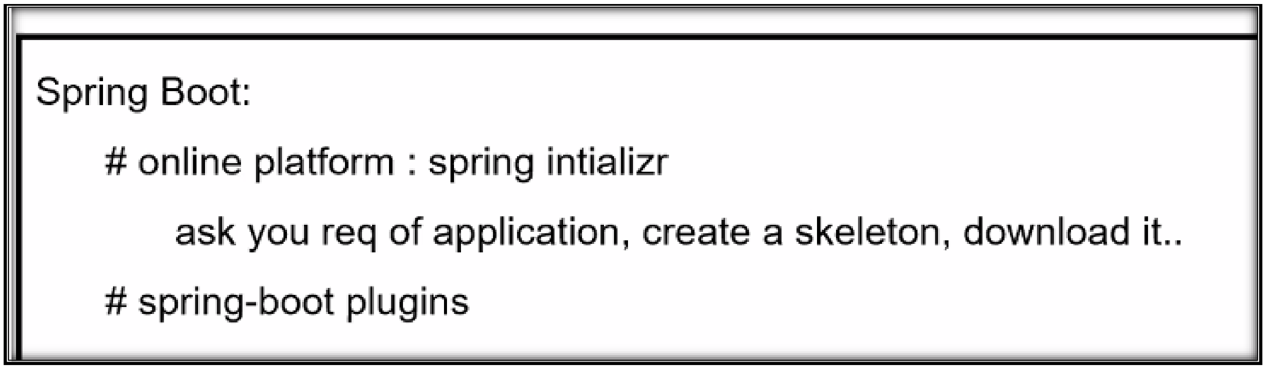


```
>java -jar <file-name>.jar
```



All requirements embedded in project

- # embed the maven cli command file
- # Text editor (no plugin dependency)
- # embedded tomcat server
- # add support of deploying app



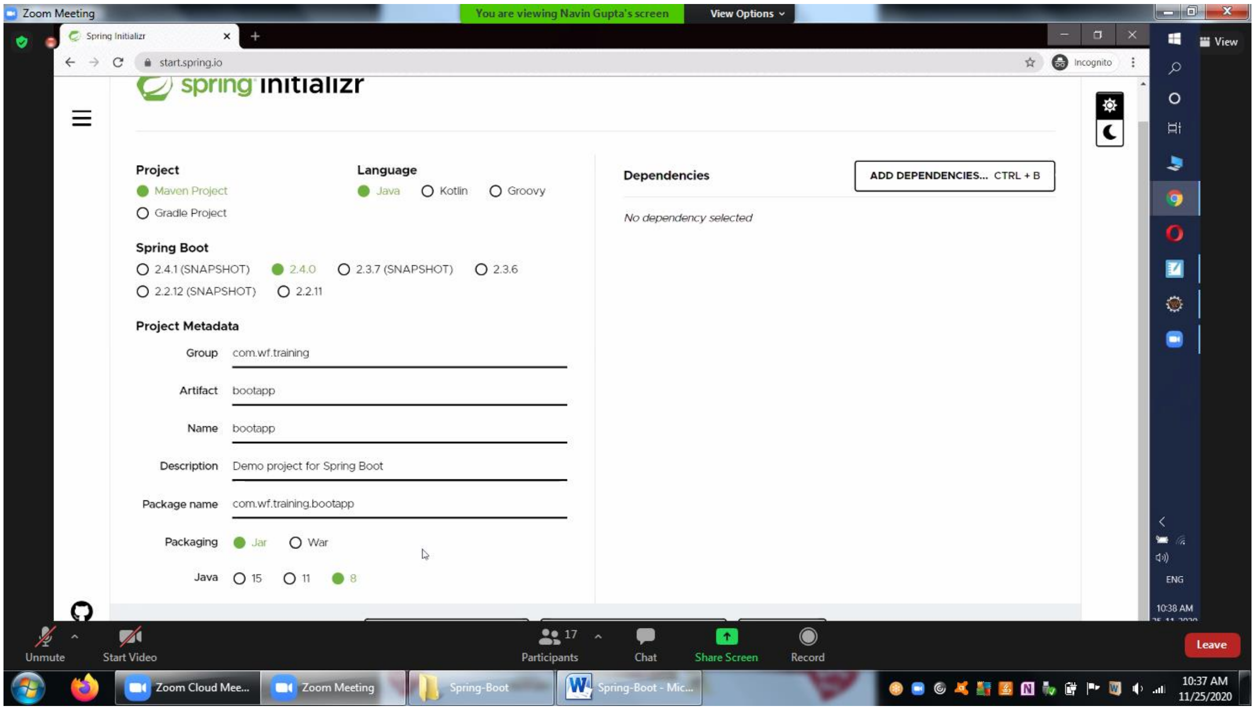
Spring Boot:

online platform : spring intializr

ask you req of application, create a skeleton, download it..

spring-boot plugins

Online Spring platform



Project
● Maven Project
○ Gradle Project

Language
● Java ○ Kotlin ○ Groovy

Spring Boot
○ 2.4.1 (SNAPSHOT) ● 2.4.0 ○ 2.3.7 (SNAPSHOT) ○ 2.3.6
○ 2.2.12 (SNAPSHOT) ○ 2.2.11

Project Metadata

Group com.wf.training

Artifact bootapp

Name bootapp

Description Demo project for Spring Boot

Package name com.wf.training.bootapp

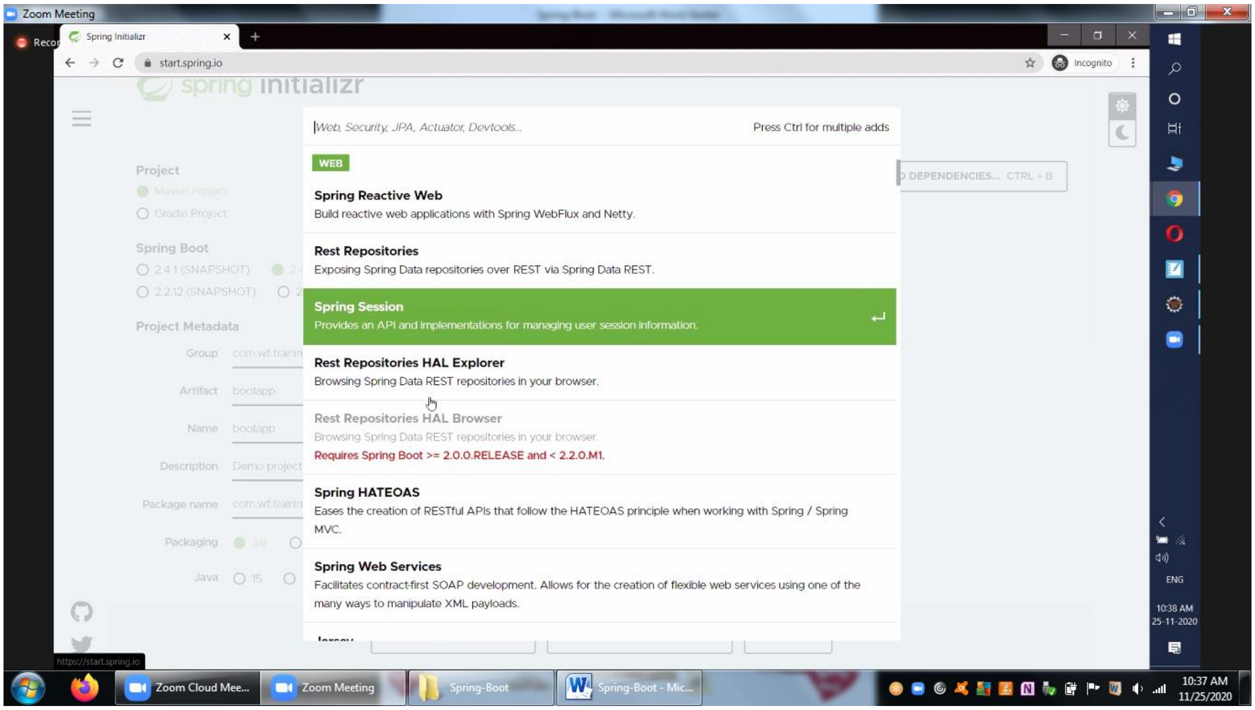
Packaging ● Jar ○ War

Java ○ 15 ○ 11 ● 8

Dependencies ADD DEPENDENCIES... CTRL + B

No dependency selected

Click on the Add Dependencies



Web, Security, JPA, Actuator, Devtools... Press Ctrl for multiple adds

WEB

Spring Reactive Web
Build reactive web applications with Spring WebFlux and Netty.

Rest Repositories
Exposing Spring Data repositories over REST via Spring Data REST.

Spring Session
Provides an API and implementations for managing user session information.

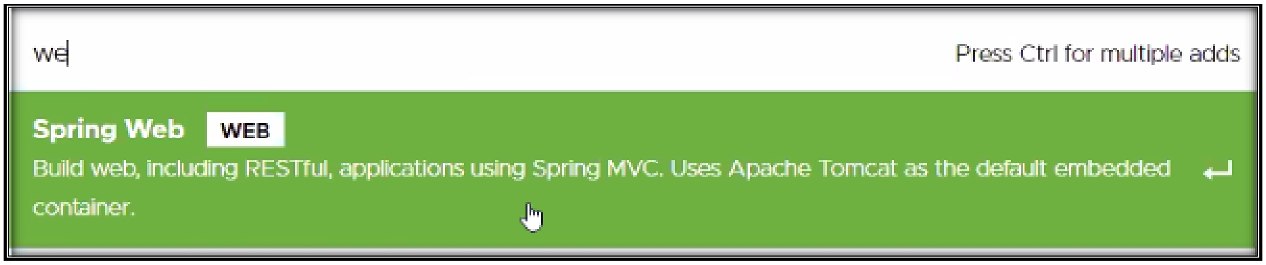
Rest Repositories HAL Explorer
Browsing Spring Data REST repositories in your browser.

Rest Repositories HAL Browser
Browsing Spring Data REST repositories in your browser.
Requires Spring Boot >= 2.0.0.RELEASE and < 2.2.0.M1.

Spring HATEOAS
Eases the creation of RESTful APIs that follow the HATEOAS principle when working with Spring / Spring MVC.

Spring Web Services
Facilitates contract-first SOAP development. Allows for the creation of flexible web services using one of the many ways to manipulate XML payloads.

Select Web project and add as dependency

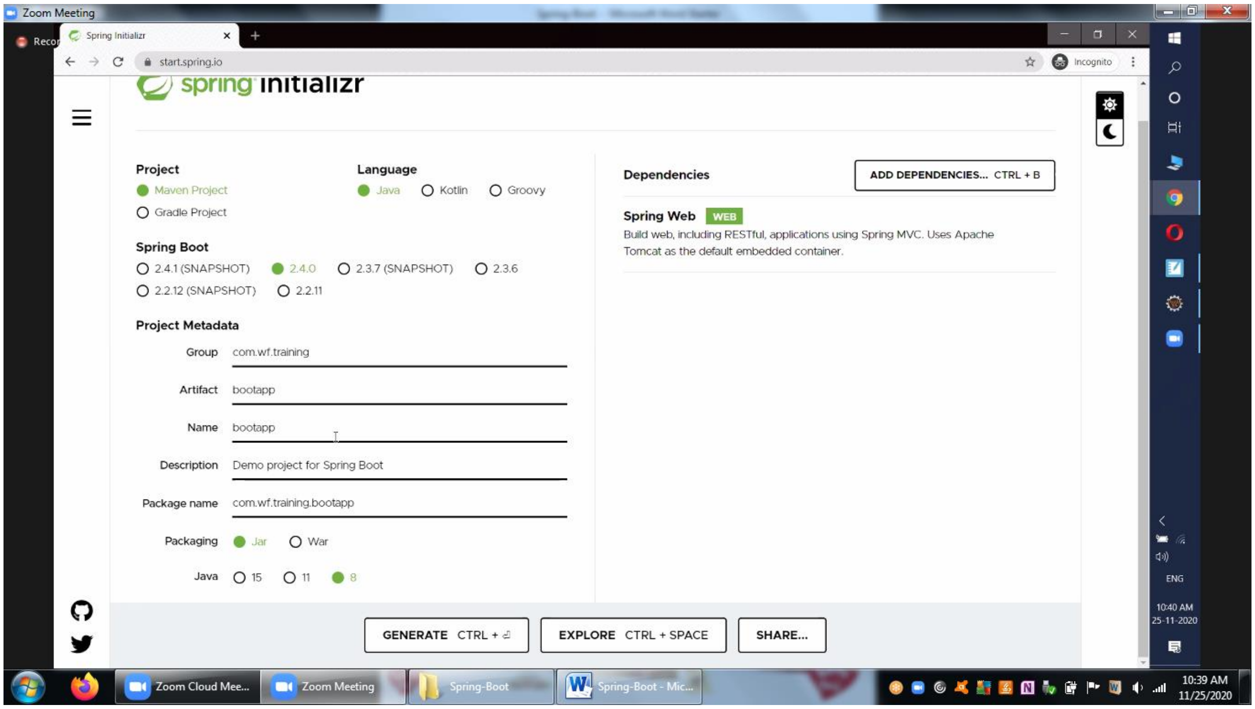


we

Press Ctrl for multiple adds

Spring Web WEB

Build web, including RESTful, applications using Spring MVC. Uses Apache Tomcat as the default embedded container.



Project
● Maven Project
○ Gradle Project

Language
● Java ○ Kotlin ○ Groovy

Spring Boot
○ 2.4.1 (SNAPSHOT) ● 2.4.0 ○ 2.3.7 (SNAPSHOT) ○ 2.3.6
○ 2.2.12 (SNAPSHOT) ○ 2.2.11

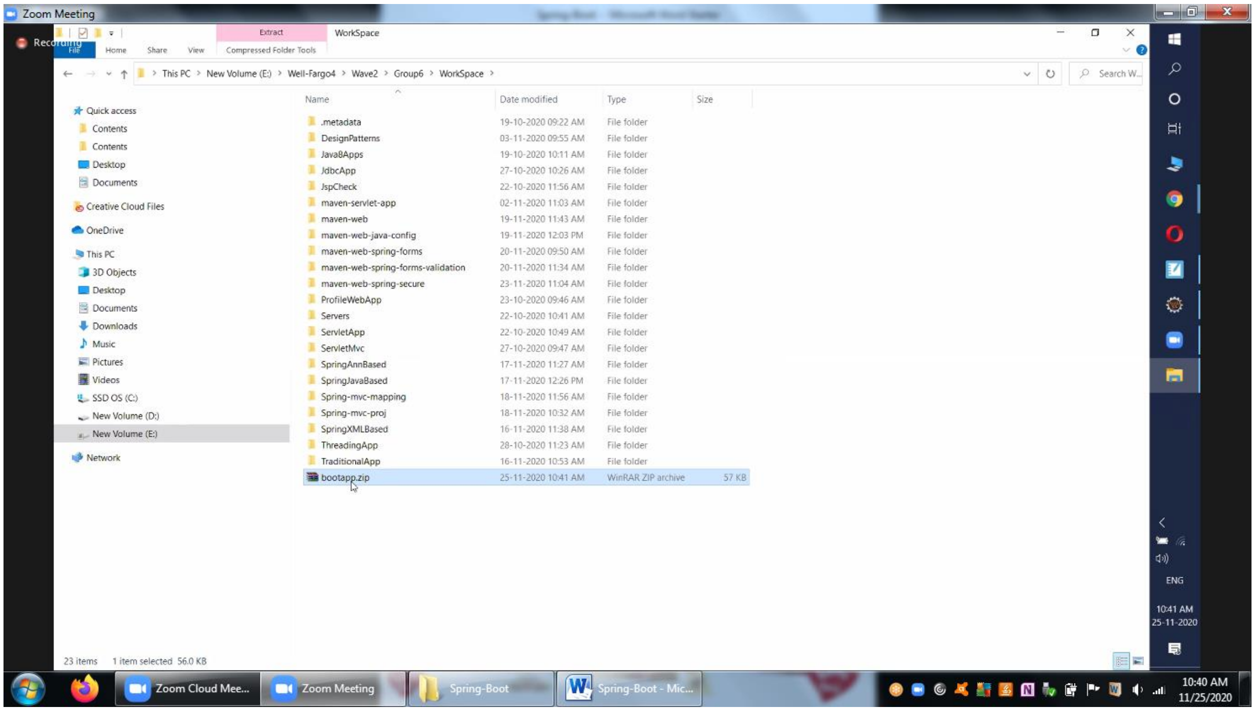
Project Metadata

Group: com.wf.training
Artifact: bootapp
Name: bootapp
Description: Demo project for Spring Boot
Package name: com.wf.training.bootapp
Packaging: ● Jar ○ War
Java: ○ 15 ○ 11 ● 8

Dependencies

Spring Web WEB
Build web, including RESTful, applications using Spring MVC. Uses Apache Tomcat as the default embedded container.

Click on the Generate CTRL and Place it in the workspace as below

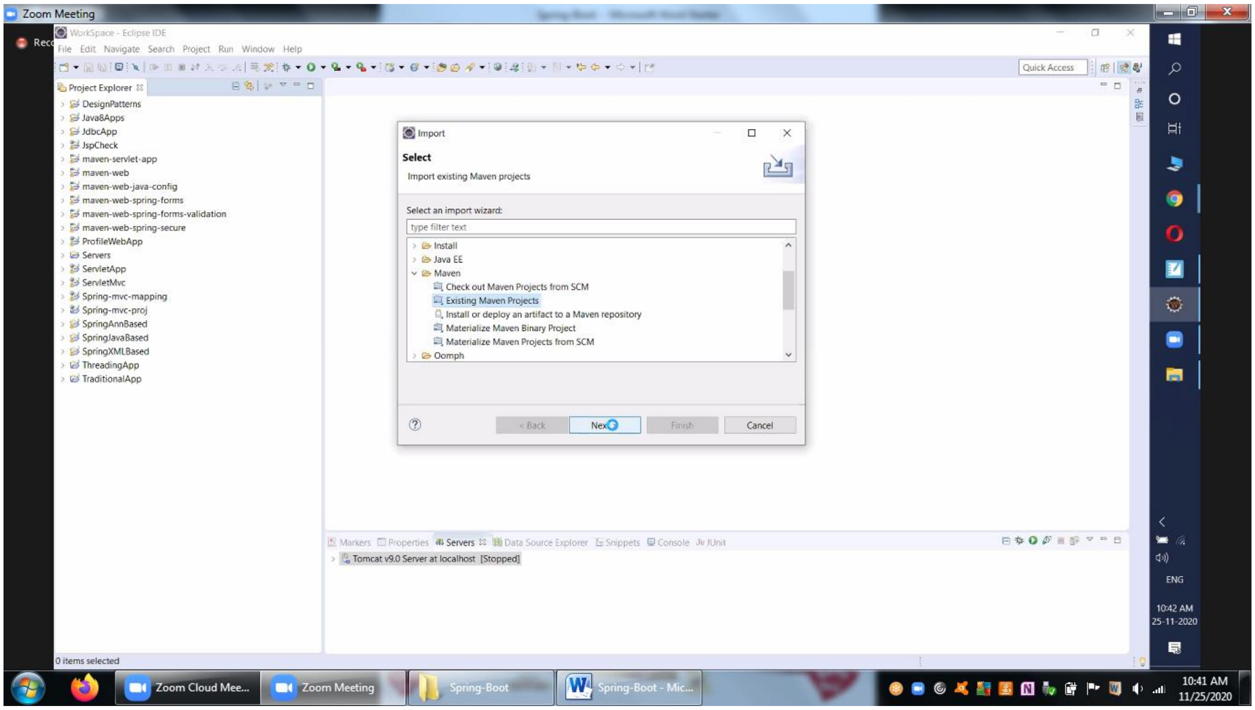


This PC > New Volume (E:) > Well-Fargo4 > Wave2 > Group6 > WorkSpace

Name	Date modified	Type	Size
.metadata	19-10-2020 09:22 AM	File folder	
DesignPatterns	03-11-2020 09:55 AM	File folder	
Java8Apps	19-10-2020 10:11 AM	File folder	
JdbcApp	27-10-2020 10:26 AM	File folder	
JspCheck	22-10-2020 11:56 AM	File folder	
maven-servlet-app	02-11-2020 11:03 AM	File folder	
maven-web	19-11-2020 11:43 AM	File folder	
maven-web-java-config	19-11-2020 12:03 PM	File folder	
maven-web-spring-forms	20-11-2020 09:50 AM	File folder	
maven-web-spring-forms-validation	20-11-2020 11:34 AM	File folder	
maven-web-spring-secure	23-11-2020 11:04 AM	File folder	
ProfileWebApp	23-10-2020 09:46 AM	File folder	
Servers	22-10-2020 10:41 AM	File folder	
ServletApp	22-10-2020 10:49 AM	File folder	
ServletMvc	27-10-2020 09:47 AM	File folder	
SpringAnnBased	17-11-2020 11:27 AM	File folder	
SpringJavaBased	17-11-2020 12:26 PM	File folder	
Spring-mvc-mapping	18-11-2020 11:56 AM	File folder	
Spring-mvc-proj	18-11-2020 10:32 AM	File folder	
SpringXMLBased	16-11-2020 11:38 AM	File folder	
ThreadingApp	28-10-2020 11:23 AM	File folder	
TraditionalApp	16-11-2020 10:53 AM	File folder	
bootapp.zip	25-11-2020 10:41 AM	WinRAR ZIP archive	57 KB

23 items 1 item selected 56.0 KB

Import as Maven Project

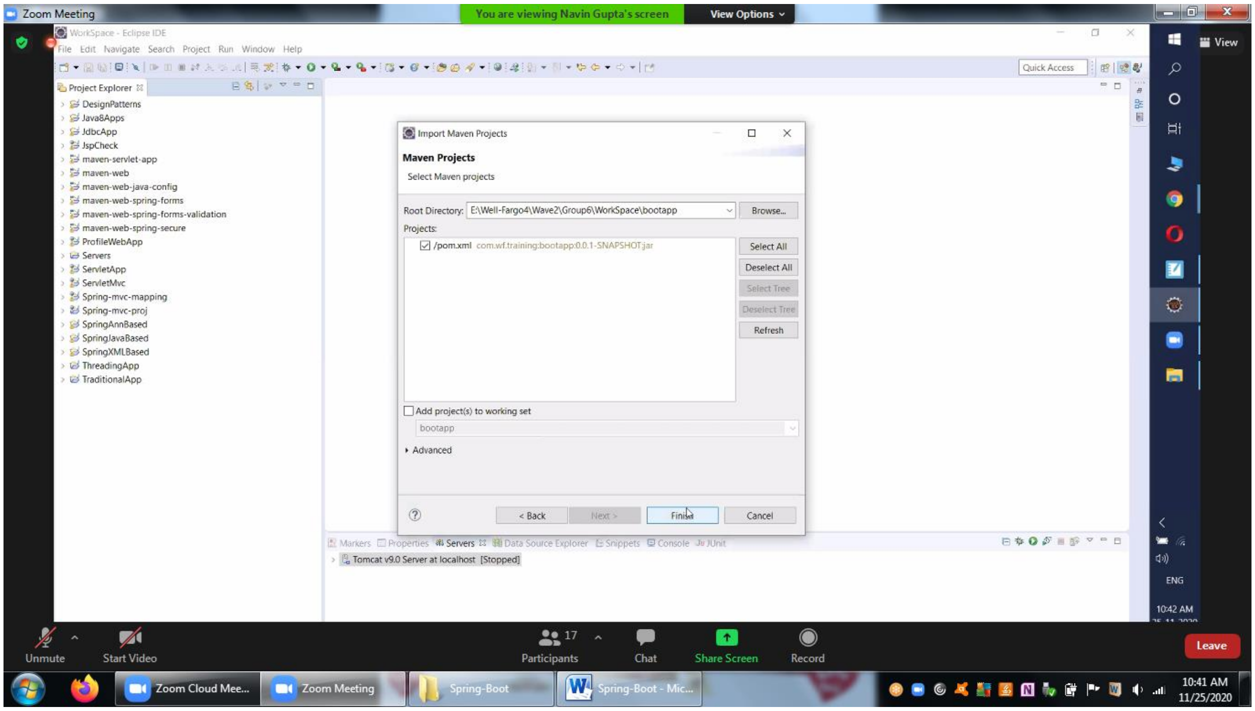


Import

Select

Import existing Maven projects

Select an import wizard:



Root Directory: E:\Well-Fargo4\Wave2\Group6\WorkSpace\bootapp

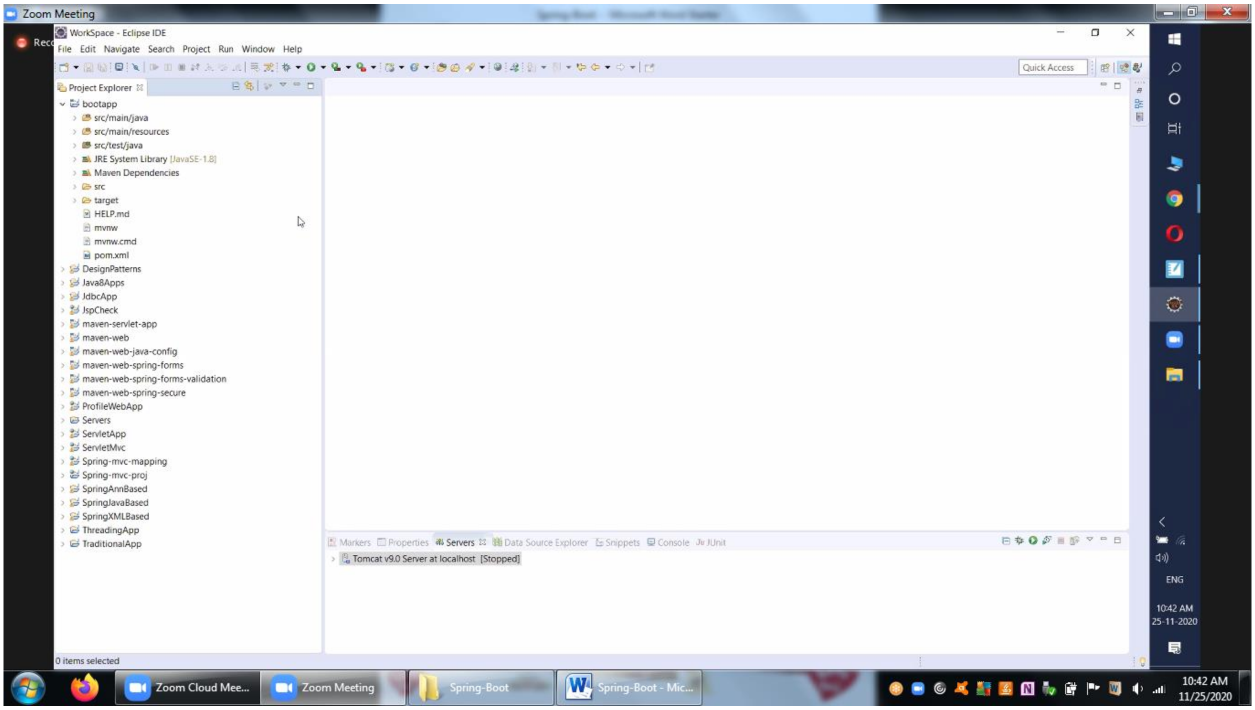
Projects:

☑ /pom.xml com.wf.training:bootapp:0.0.1-SNAPSHOT.jar

☐ Add project(s) to working set

bootapp

Tomcat v9.0 Server at localhost [Stopped]



bootapp
- src/main/java
- src/main/resources
- src/test/java
- JRE System Library [JavaSE-1.8]
- Maven Dependencies
- src
- target
- HELP.md
- mvnw
- mvnw.cmd
- pom.xml

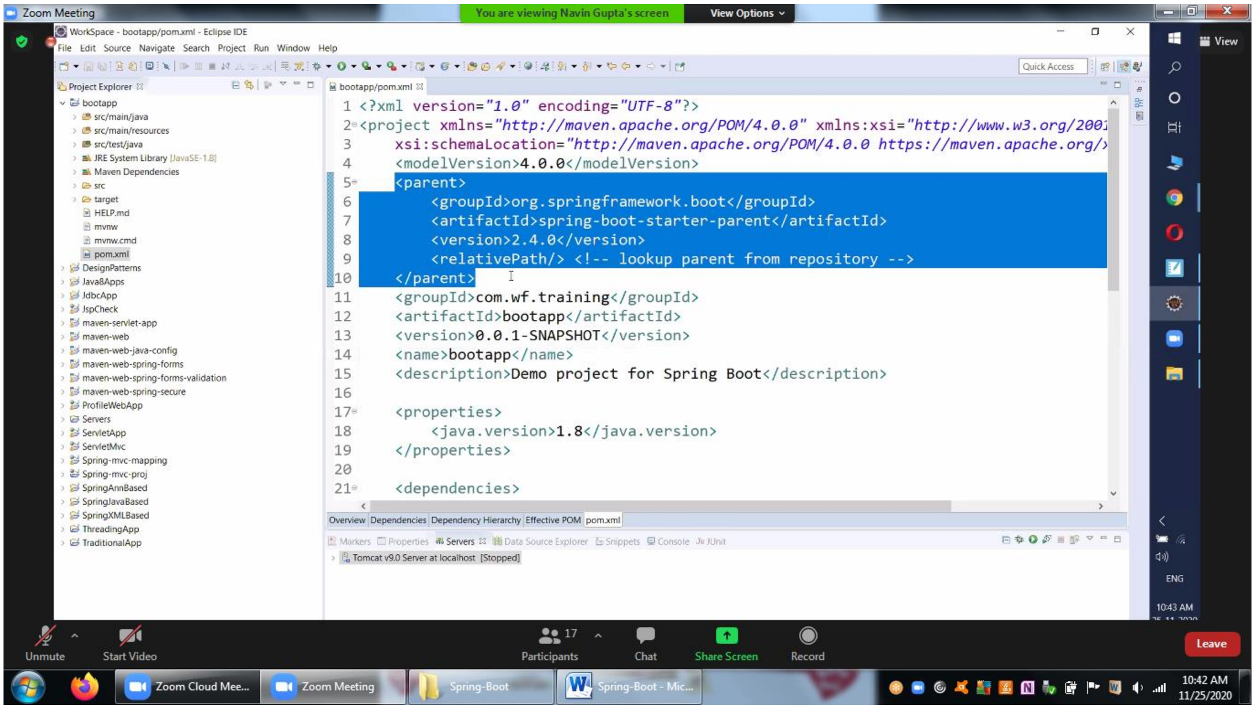
DesignPatterns
Java8Apps
JdbcApp
JspCheck
maven-servlet-app
maven-web
maven-web-java-config
maven-web-spring-forms
maven-web-spring-forms-validation
maven-web-spring-secure
ProfileWebApp
Servers
ServletApp
ServletMvc
Spring-mvc-mapping
Spring-mvc-proj
SpringAnnBased
SpringJavaBased
SpringXMLBased
ThreadingApp
TraditionalApp

Tomcat v9.0 Server at localhost [Stopped]

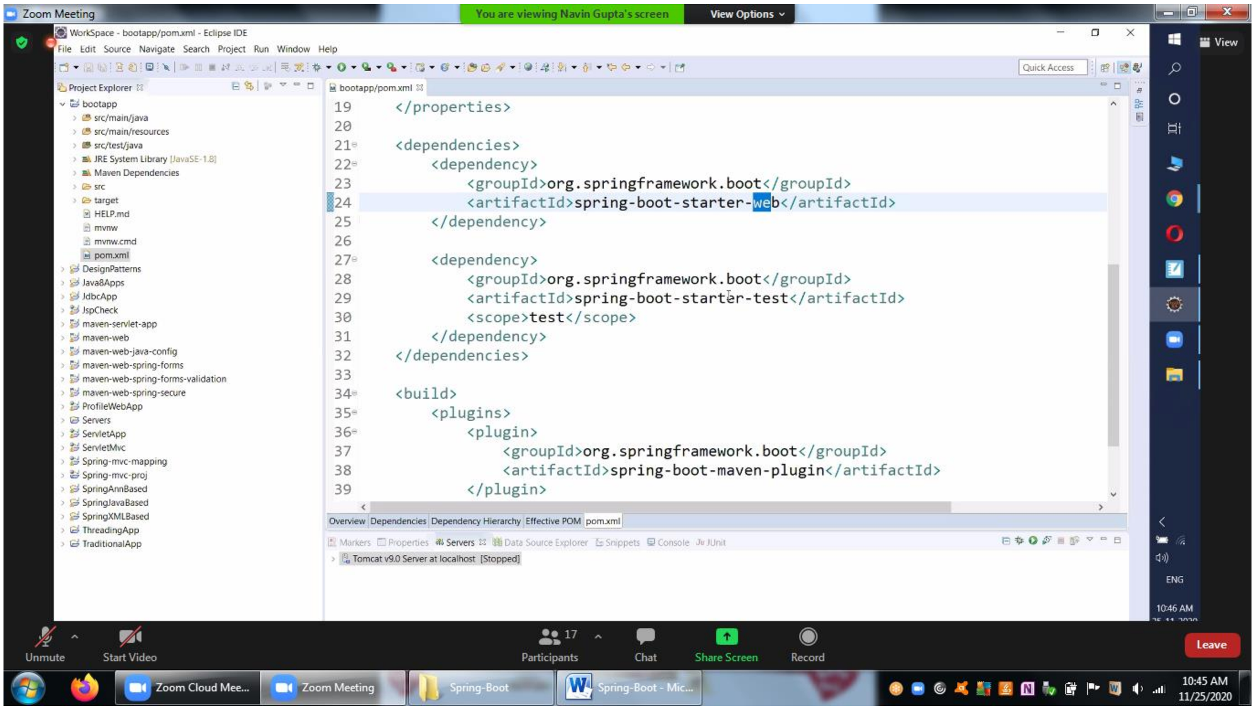
0 items selected

Dependencies Management

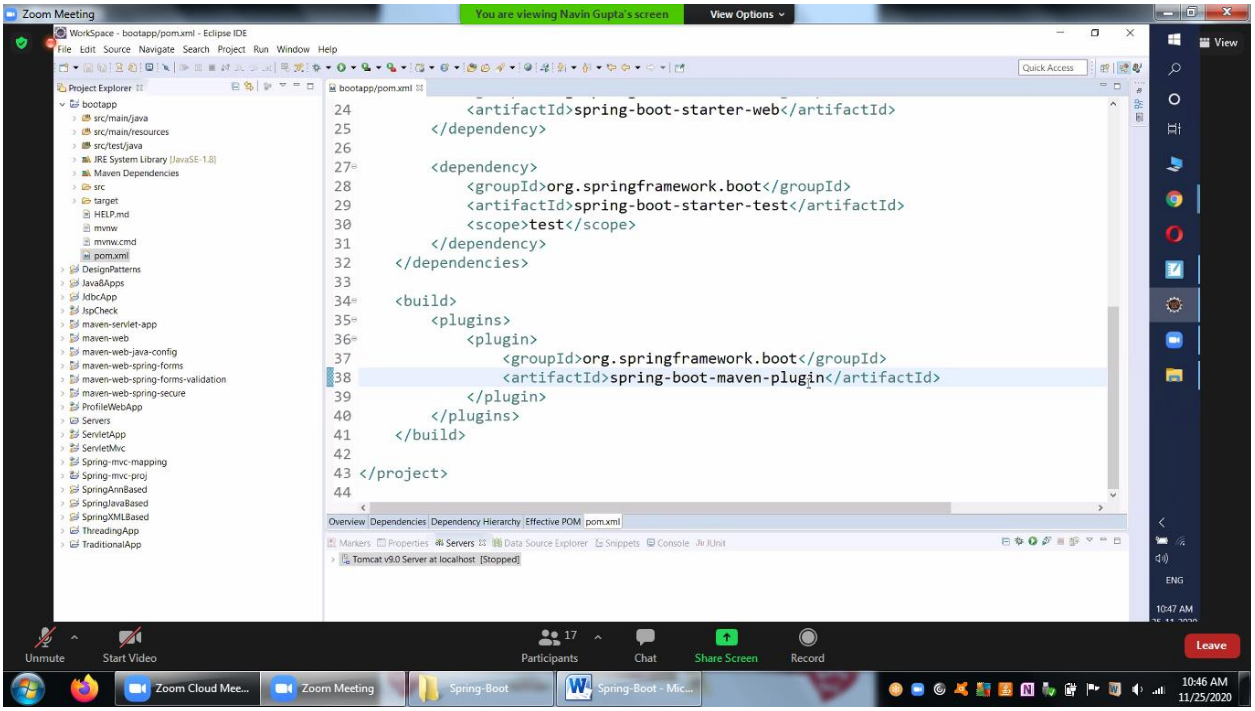
Existing Maven dependencies



```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
    xsi:schemaLocation="http://maven.apache.org/POM/4.0.0 https://maven.apache.org/x
    <modelVersion>4.0.0</modelVersion>
    <parent>
        <groupId>org.springframework.boot</groupId>
        <artifactId>spring-boot-starter-parent</artifactId>
        <version>2.4.0</version>
        <relativePath/> <!-- lookup parent from repository -->
    </parent>
    <groupId>com.wf.training</groupId>
    <artifactId>bootapp</artifactId>
    <version>0.0.1-SNAPSHOT</version>
    <name>bootapp</name>
    <description>Demo project for Spring Boot</description>

    <properties>
        <java.version>1.8</java.version>
    </properties>

    <dependencies>
```

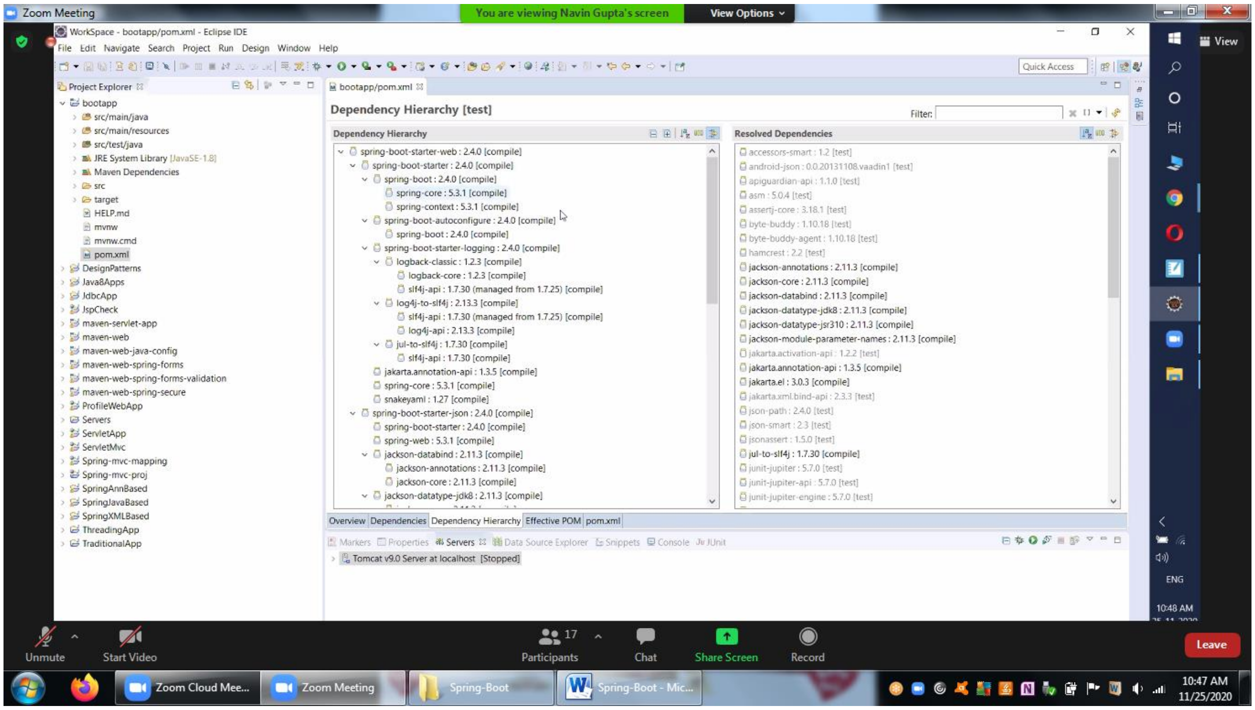

Zoom Meeting
You are viewing Navin Gupta's screen
View Options
WorkSpace - bootapp/pom.xml - Eclipse IDE
File Edit Source Navigate Search Project Run Window Help
Quick Access
Project Explorer
bootapp/pom.xml
Overview Dependencies Dependency Hierarchy Effective POM pom.xml
Markers Properties Servers Data Source Explorer Snippets Console JUnit
Tomcat v9.0 Server at localhost [Stopped]
View
ENG
10:46 AM
Unmute
Start Video
17
Participants
Chat
Share Screen
Record
Leave
Zoom Cloud Mee...
Zoom Meeting
Spring-Boot
Spring-Boot - Mic...
10:45 AM
11/25/2020

```xml
    </properties>

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


Zoom Meeting
You are viewing Navin Gupta's screen
View Options
WorkSpace - bootapp/pom.xml - Eclipse IDE
File Edit Source Navigate Search Project Run Window Help
Project Explorer
bootapp/pom.xml
Overview Dependencies Dependency Hierarchy Effective POM pom.xml
Markers Properties Servers Data Source Explorer Snippets Console JUnit
Tomcat v9.0 Server at localhost [Stopped]
Unmute Start Video Participants Chat Share Screen Record Leave

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

For Exploring you can go to Hierarchy Tab


Zoom Meeting
You are viewing Navin Gupta's screen
View Options
WorkSpace - bootapp/pom.xml - Eclipse IDE
File Edit Navigate Search Project Run Design Window Help
Project Explorer
bootapp/pom.xml
Filter:
Overview Dependencies Dependency Hierarchy Effective POM pom.xml
Markers Properties Servers Data Source Explorer Snippets Console JUnit
Tomcat v9.0 Server at localhost [Stopped]
Unmute Start Video Participants Chat Share Screen Record Leave

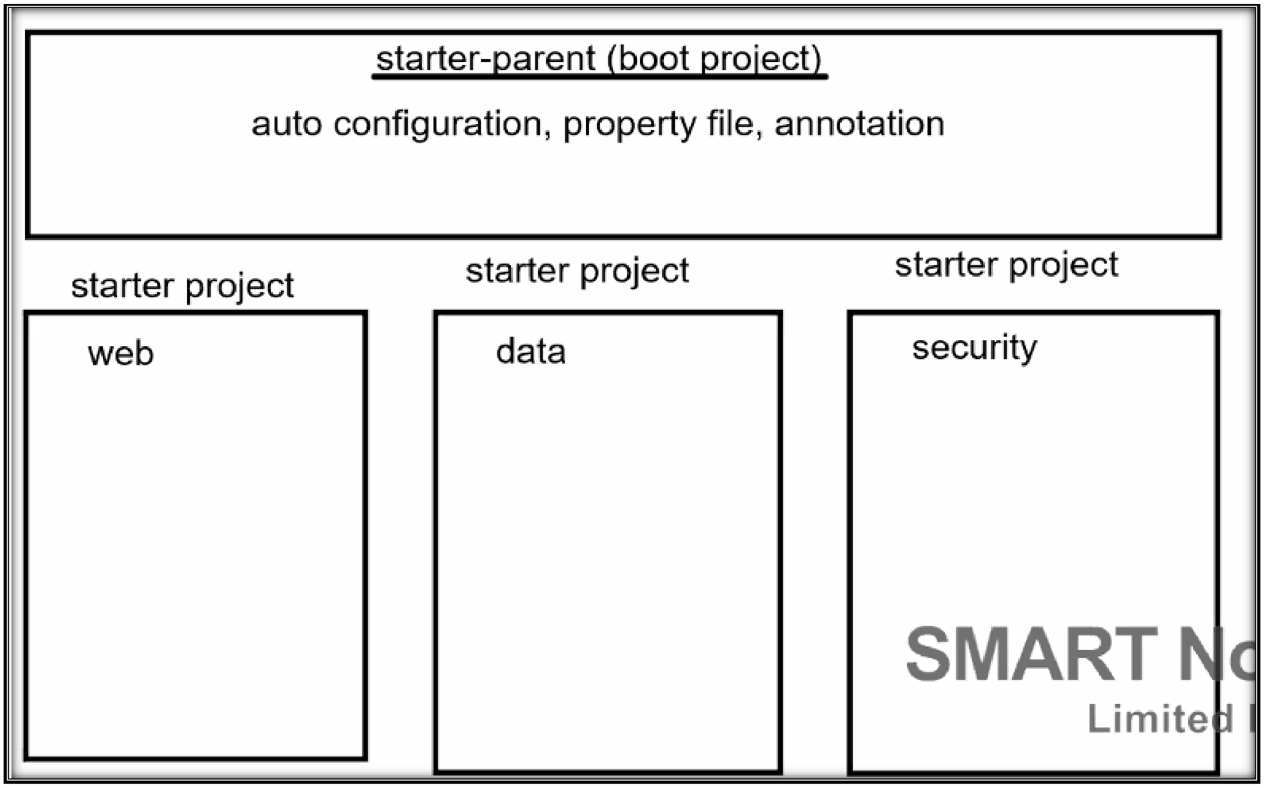
Dependency Hierarchy [test]

Dependency Hierarchy

```
spring-boot-starter-web : 2.4.0 [compile]
  spring-boot-starter : 2.4.0 [compile]
    spring-boot : 2.4.0 [compile]
      spring-core : 5.3.1 [compile]
      spring-context : 5.3.1 [compile]
    spring-boot-autoconfigure : 2.4.0 [compile]
      spring-boot : 2.4.0 [compile]
    spring-boot-starter-logging : 2.4.0 [compile]
      logback-classic : 1.2.3 [compile]
        logback-core : 1.2.3 [compile]
        slf4j-api : 1.7.30 (managed from 1.7.25) [compile]
      log4j-to-slf4j : 2.13.3 [compile]
        slf4j-api : 1.7.30 (managed from 1.7.25) [compile]
        log4j-api : 2.13.3 [compile]
      jul-to-slf4j : 1.7.30 [compile]
        slf4j-api : 1.7.30 [compile]
    jakarta.annotation-api : 1.3.5 [compile]
    spring-core : 5.3.1 [compile]
    snakeyaml : 1.27 [compile]
  spring-boot-starter-json : 2.4.0 [compile]
    spring-boot-starter : 2.4.0 [compile]
    spring-web : 5.3.1 [compile]
    jackson-databind : 2.11.3 [compile]
      jackson-annotations : 2.11.3 [compile]
      jackson-core : 2.11.3 [compile]
    jackson-datatype-jdk8 : 2.11.3 [compile]
```

Resolved Dependencies

```
accessors-smart : 1.2 [test]
android-json : 0.0.20131108.vaadin1 [test]
apiguardian-api : 1.1.0 [test]
asm : 5.0.4 [test]
assertj-core : 3.18.1 [test]
byte-buddy : 1.10.18 [test]
byte-buddy-agent : 1.10.18 [test]
hamcrest : 2.2 [test]
jackson-annotations : 2.11.3 [compile]
jackson-core : 2.11.3 [compile]
jackson-databind : 2.11.3 [compile]
jackson-datatype-jdk8 : 2.11.3 [compile]
jackson-datatype-jsr310 : 2.11.3 [compile]
jackson-module-parameter-names : 2.11.3 [compile]
jakarta.activation-api : 1.2.2 [test]
jakarta.annotation-api : 1.3.5 [compile]
jakarta.el : 3.0.3 [compile]
jakarta.xml.bind-api : 2.3.3 [test]
json-path : 2.4.0 [test]
json-smart : 2.3 [test]
jsonassert : 1.5.0 [test]
jul-to-slf4j : 1.7.30 [compile]
junit-jupiter : 5.7.0 [test]
junit-jupiter-api : 5.7.0 [test]
junit-jupiter-engine : 5.7.0 [test]
```



starter-parent (boot project)

auto configuration, property file, annotation

starter project

web

starter project

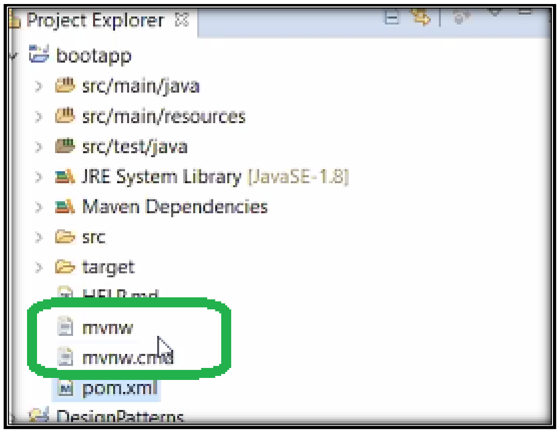
data

starter project

security

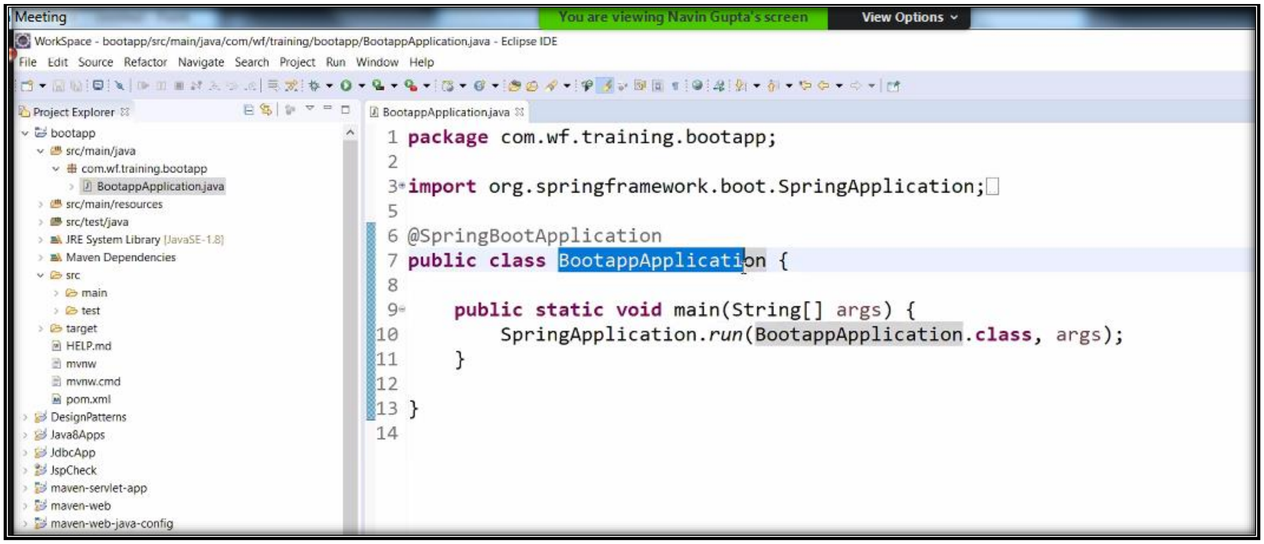
Let’s talk about Self-Sufficiency

Maven in-built support is below



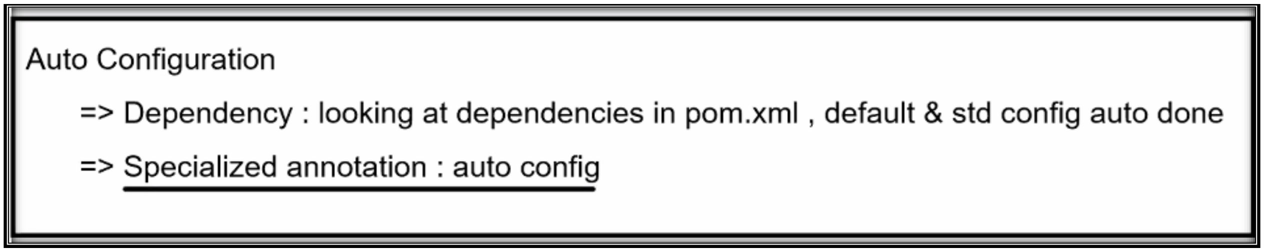
```
bootapp
  src/main/java
  src/main/resources
  src/test/java
  JRE System Library [JavaSE-1.8]
  Maven Dependencies
  src
  target
  HELP.md
  mvnw
  mvnw.cmd
  pom.xml
DesignPatterns
```

In built Java class along with configuration is already given



```
package com.wf.training.bootapp;

import org.springframework.boot.SpringApplication;

@SpringBootApplication
public class BootappApplication {

    public static void main(String[] args) {
        SpringApplication.run(BootappApplication.class, args);
    }

}
```

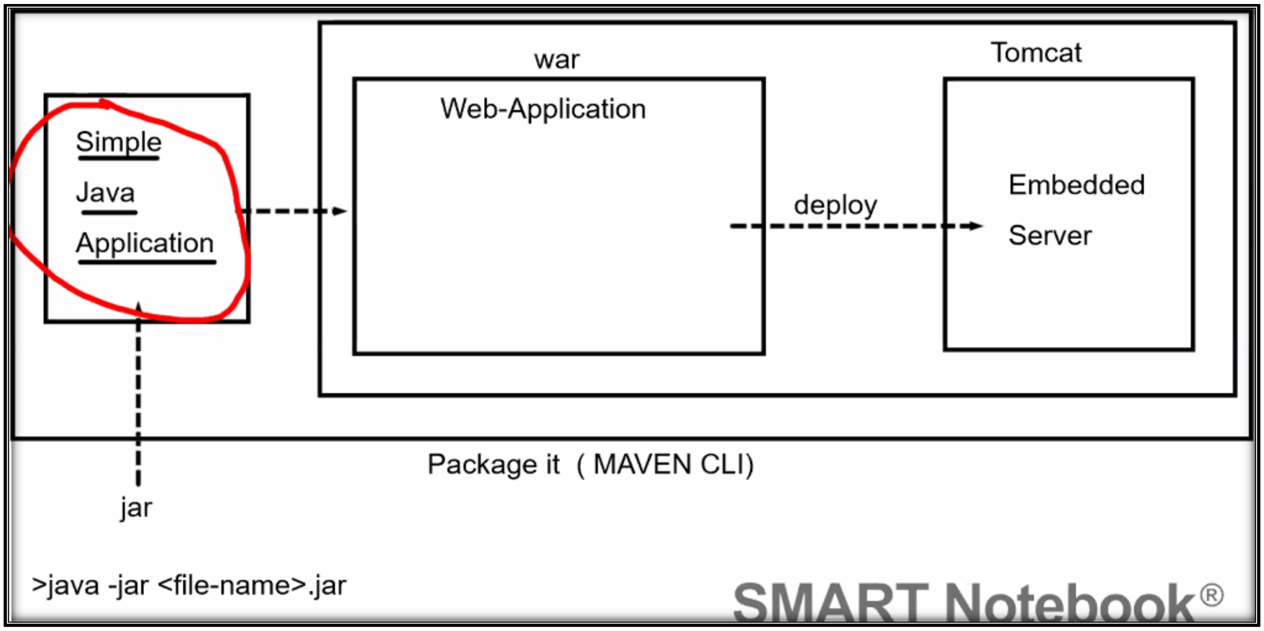
Explanation



Auto Configuration

=> Dependency : looking at dependencies in pom.xml , default & std config auto done

=> Specialized annotation : auto config



Simple Java Application

war

Web-Application

deploy

Tomcat

Embedded Server

Package it (MAVEN CLI)

jar

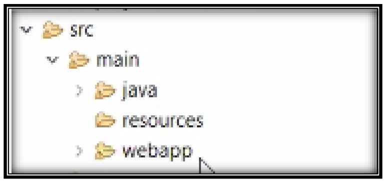
>java -jar <file-name>.jar

Just we have to focus on the Views, Model and Controllers (Business resources)

Create the Spring Resources which are specific for our project

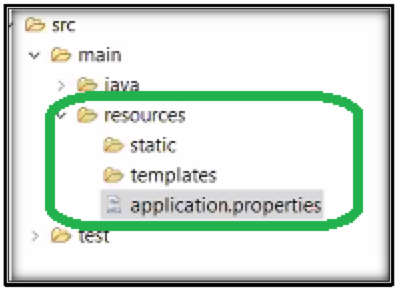
Webapp folder

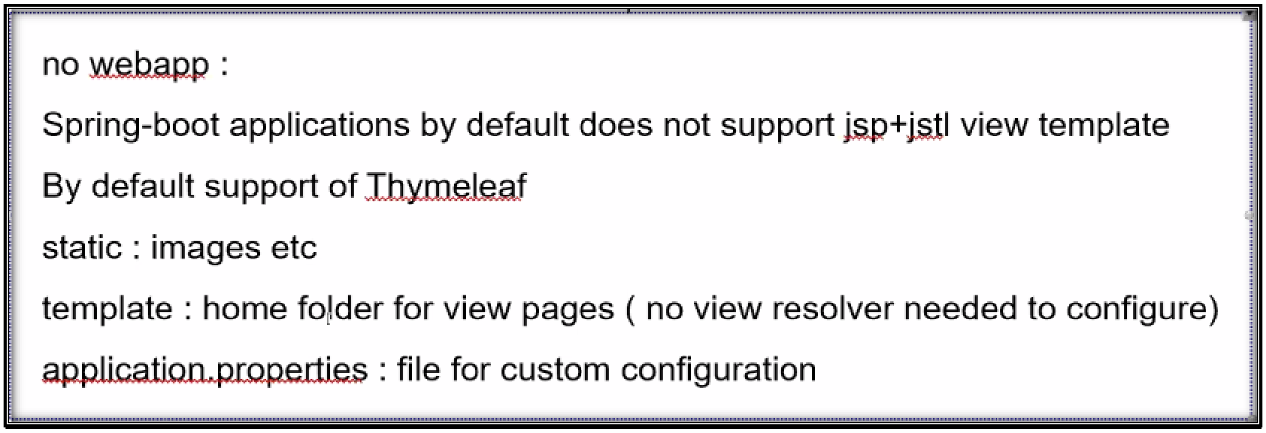
webapp folder is missing and it is expected, we don’t see the webapp folder for spring boot



Explanation:

We see new resoruces 3 folders and explanation is below





no webapp :

Spring-boot applications by default does not support jsp+jstl view template

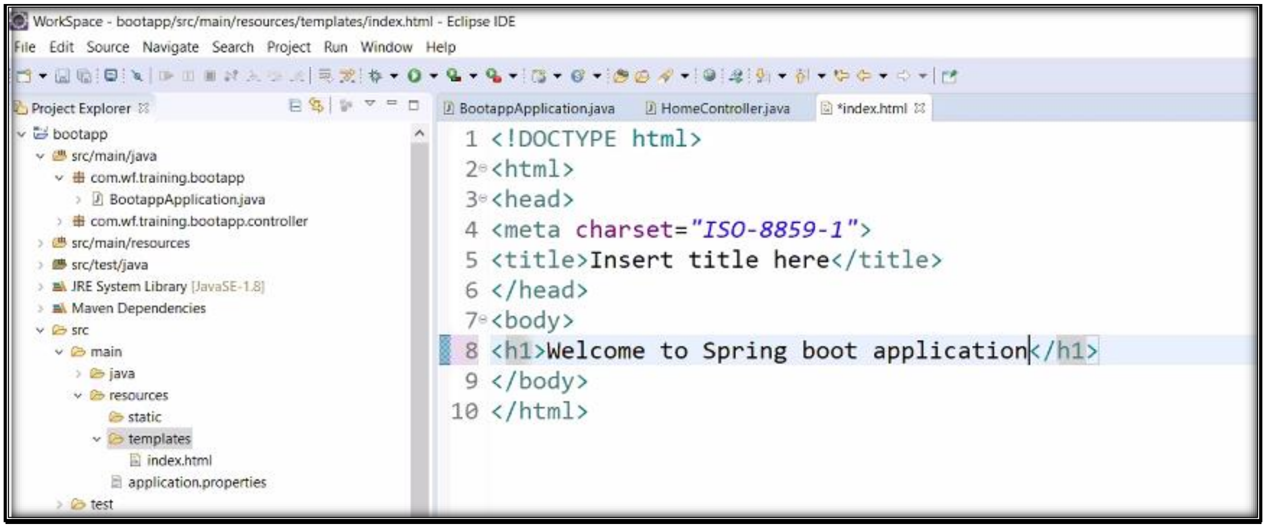
By default support of Thymeleaf

static : images etc

template : home folder for view pages (no view resolver needed to configure)

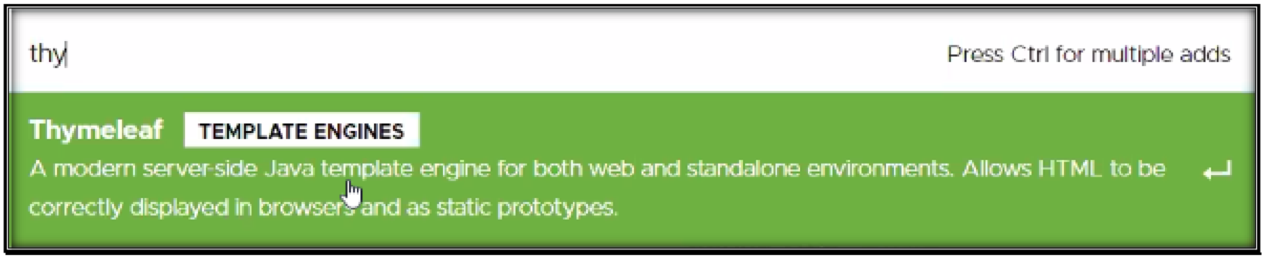
application.properties : file for custom configuration

Add index.html for basic test scenario



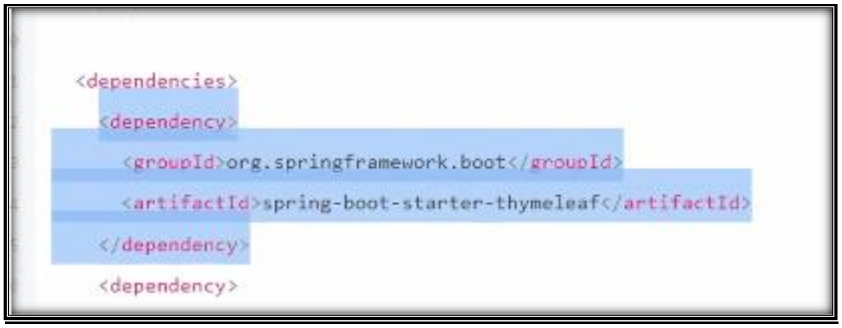
```html
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Insert title here</title>
</head>
<body>
<h1>Welcome to Spring boot application</h1>
</body>
</html>
```

Add the dependency of ThymeLeaf

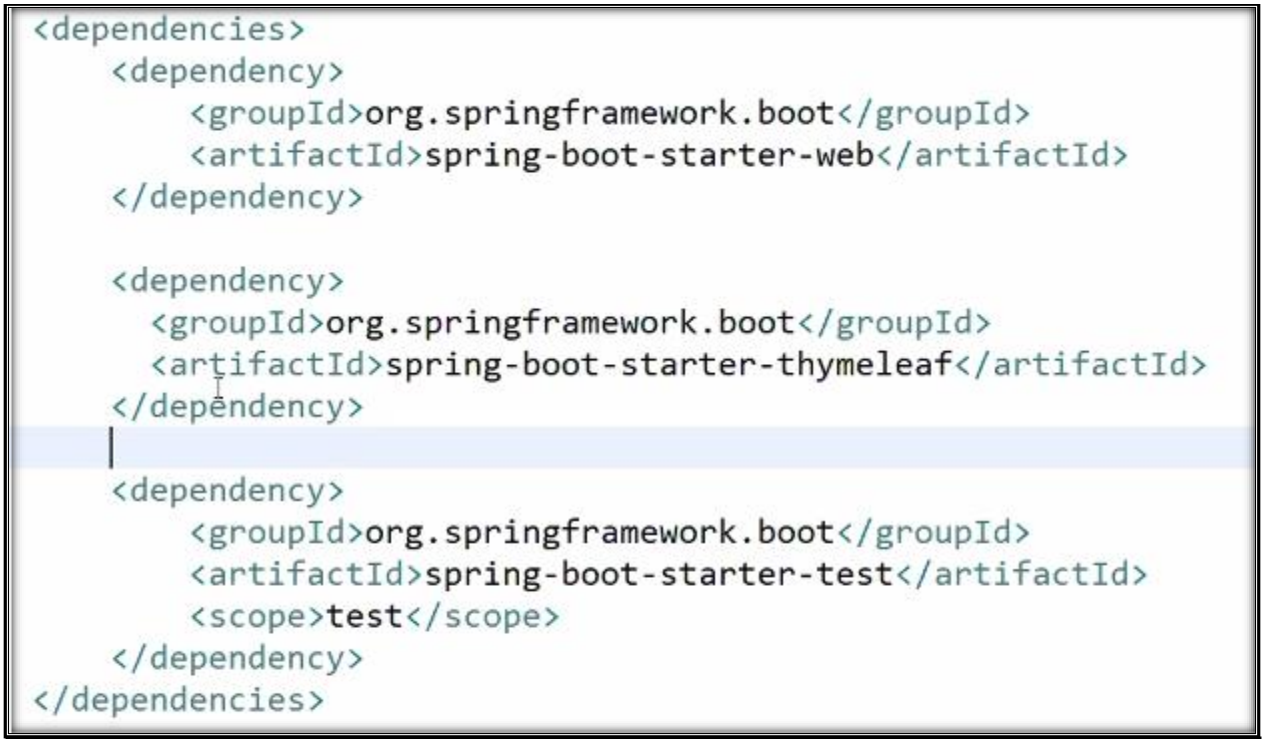


A modern server-side Java template engine for both web and standalone environments. Allows HTML to be correctly displayed in browsers and as static prototypes.

Click on the Explore button, Copy and add the same to our project POM.xml

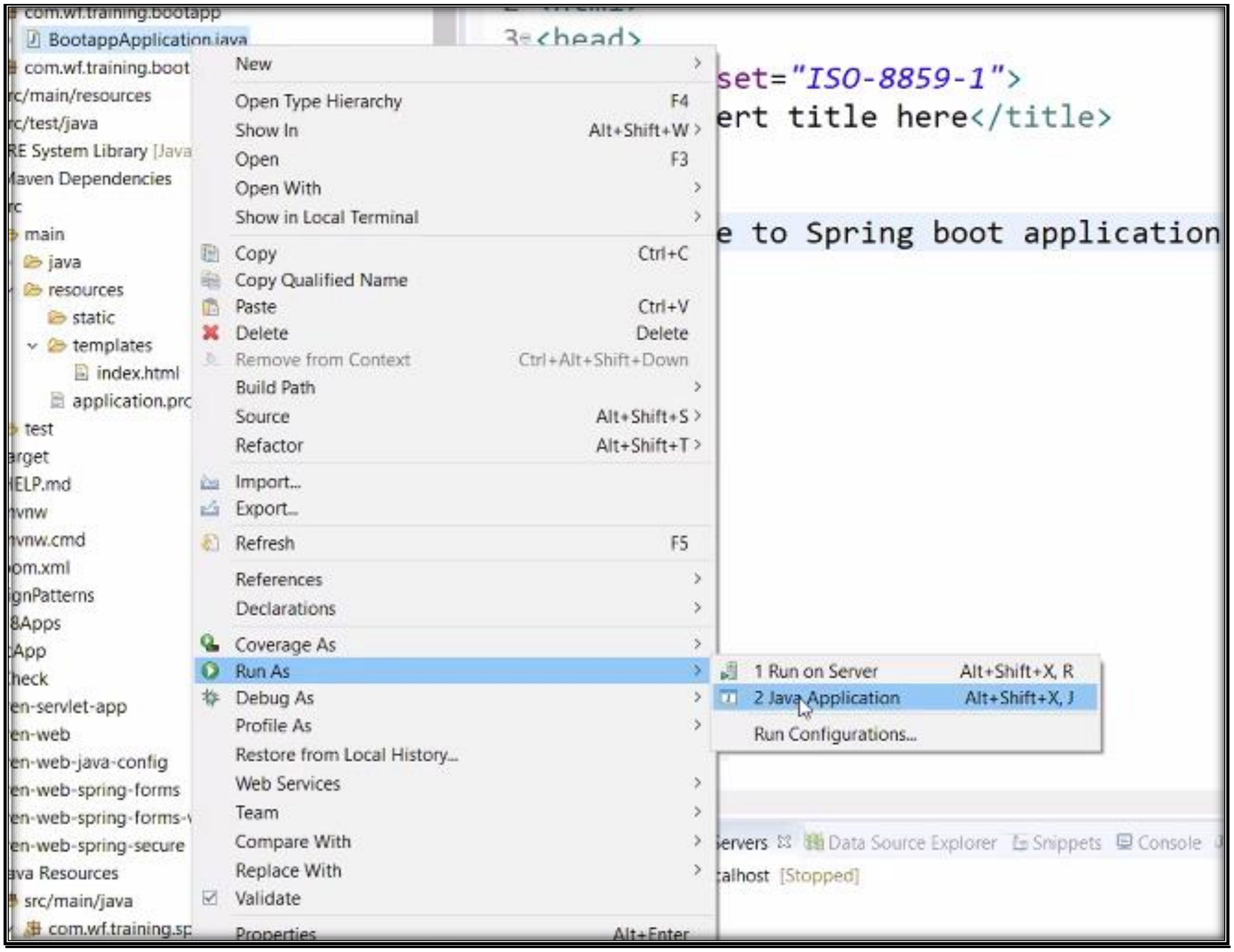


```xml
<dependencies>
  <dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-thymeleaf</artifactId>
  </dependency>
  <dependency>
```

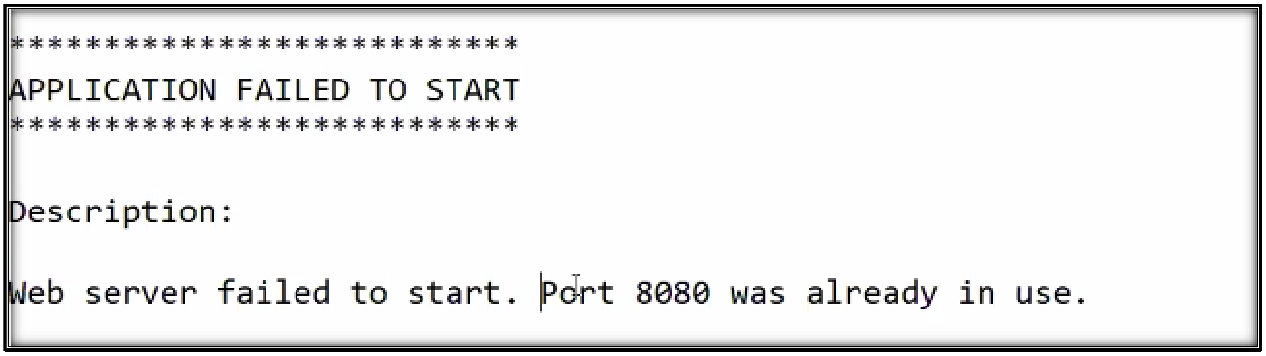


```
<dependencies>
    <dependency>
        <groupId>org.springframework.boot</groupId>
        <artifactId>spring-boot-starter-web</artifactId>
    </dependency>

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

    <dependency>
        <groupId>org.springframework.boot</groupId>
        <artifactId>spring-boot-starter-test</artifactId>
        <scope>test</scope>
    </dependency>
</dependencies>
```

Run our Application and Run as Java application



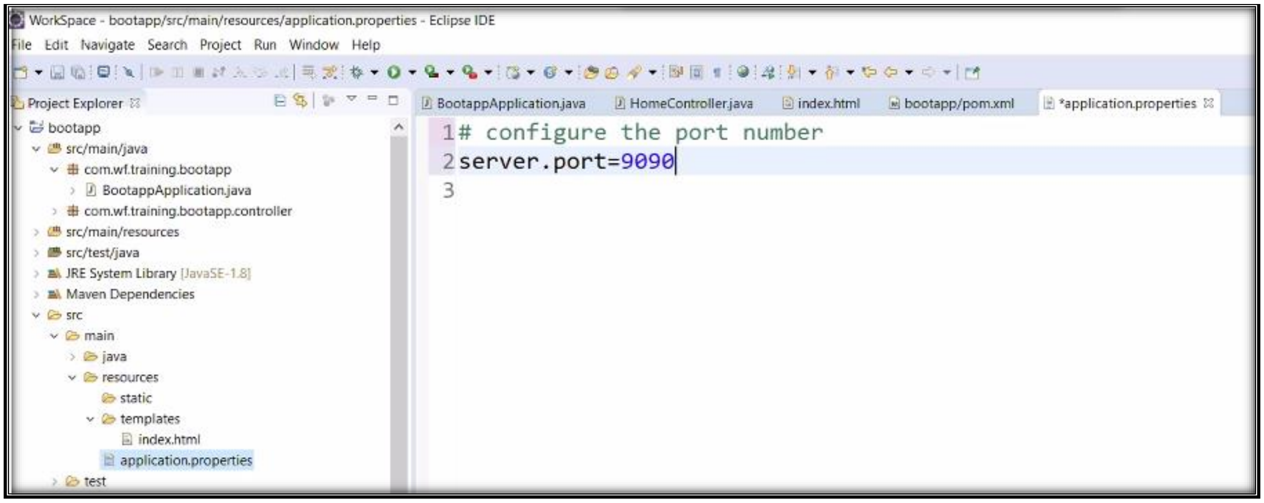
We may get this error



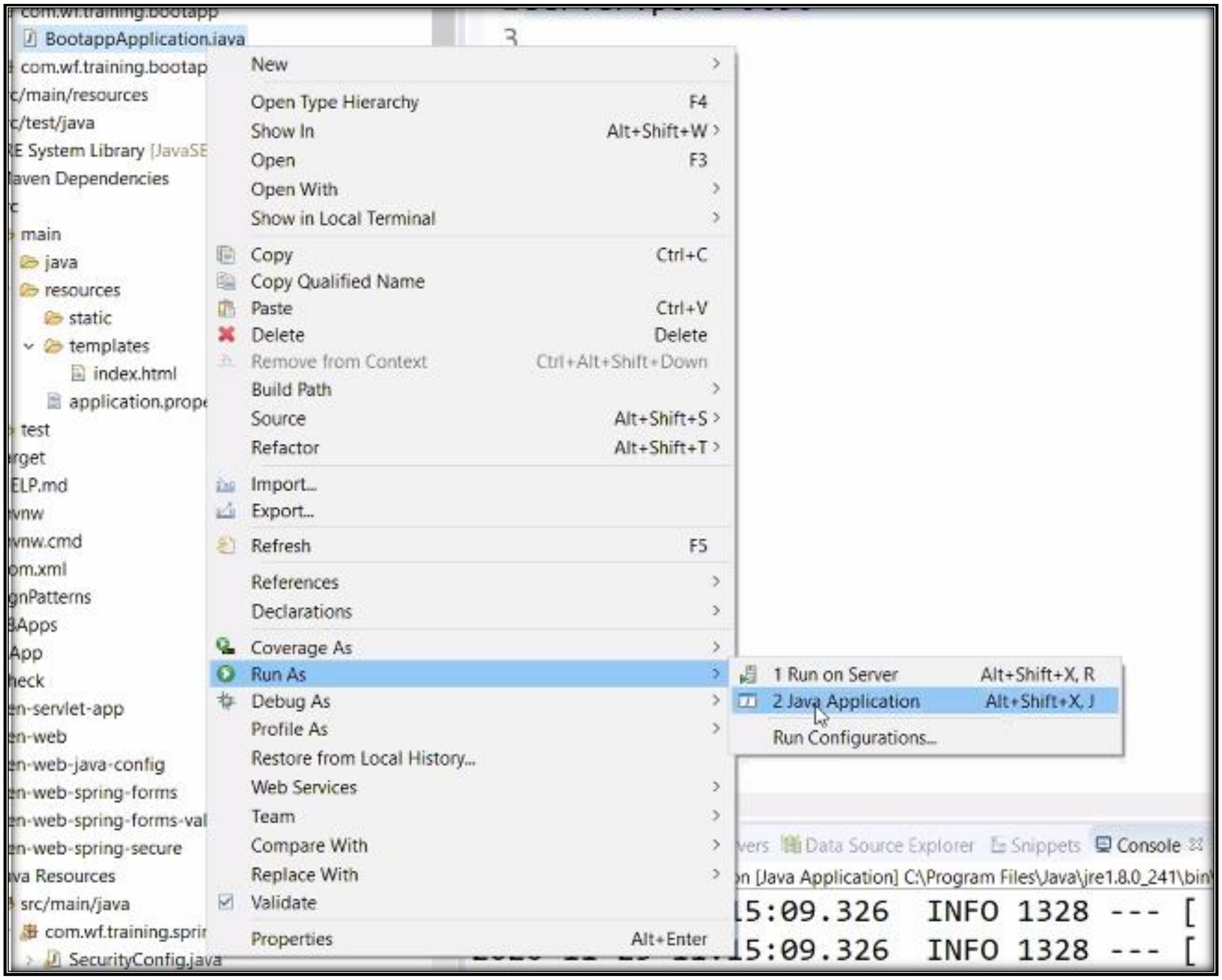
```
***************************
APPLICATION FAILED TO START
***************************


Description:

Web server failed to start. Port 8080 was already in use.
```

Configure the port number for the embedded Tomcat

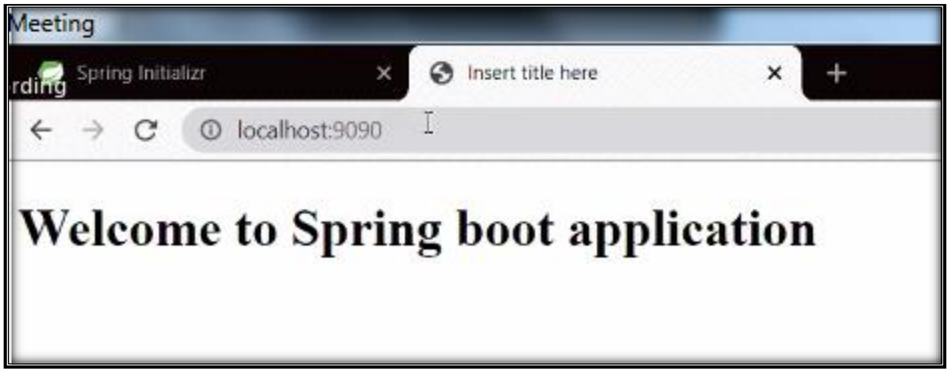
Here application.properties plays a vital role, add the below key value pair in properties file



```
# configure the port number
server.port=9090
```



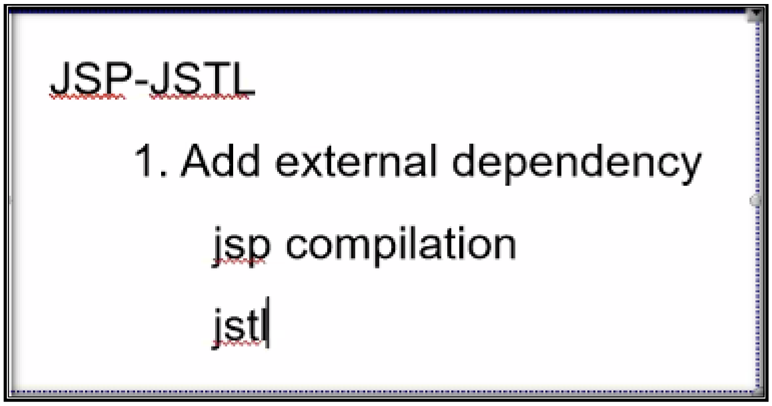
We have to enter the URL and launch the chrome manually. Now its success.



Welcome to Spring boot application

....Basic Configuration is success and completed now....

Adding JSP-JSTL dependencies from mvvrepository

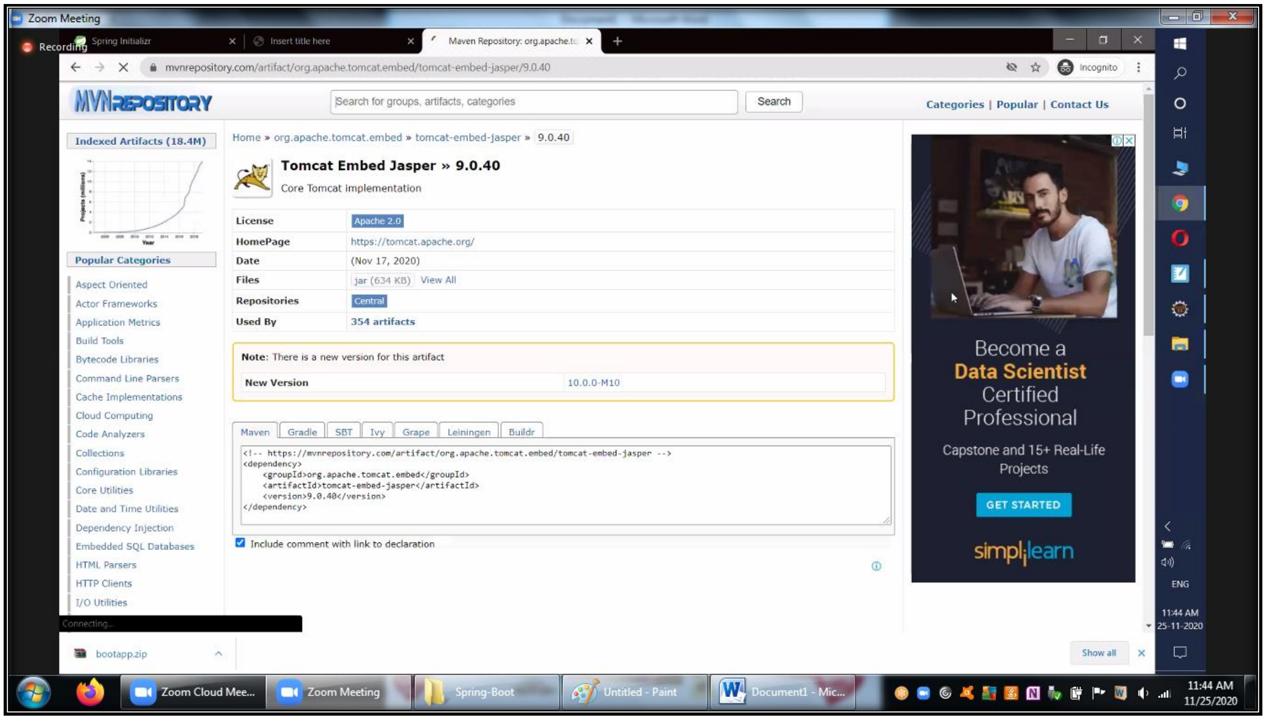


JSP-JSTL

1. Add external dependency

jsp compilation

jstl



Home » org.apache.tomcat.embed » tomcat-embed-jasper » 9.0.40

Tomcat Embed Jasper » 9.0.40

Core Tomcat implementation

License	Apache 2.0
HomePage	https://tomcat.apache.org/
Date	(Nov 17, 2020)
Files	jar (634 KB) View All
Repositories	Central
Used By	354 artifacts

Note: There is a new version for this artifact

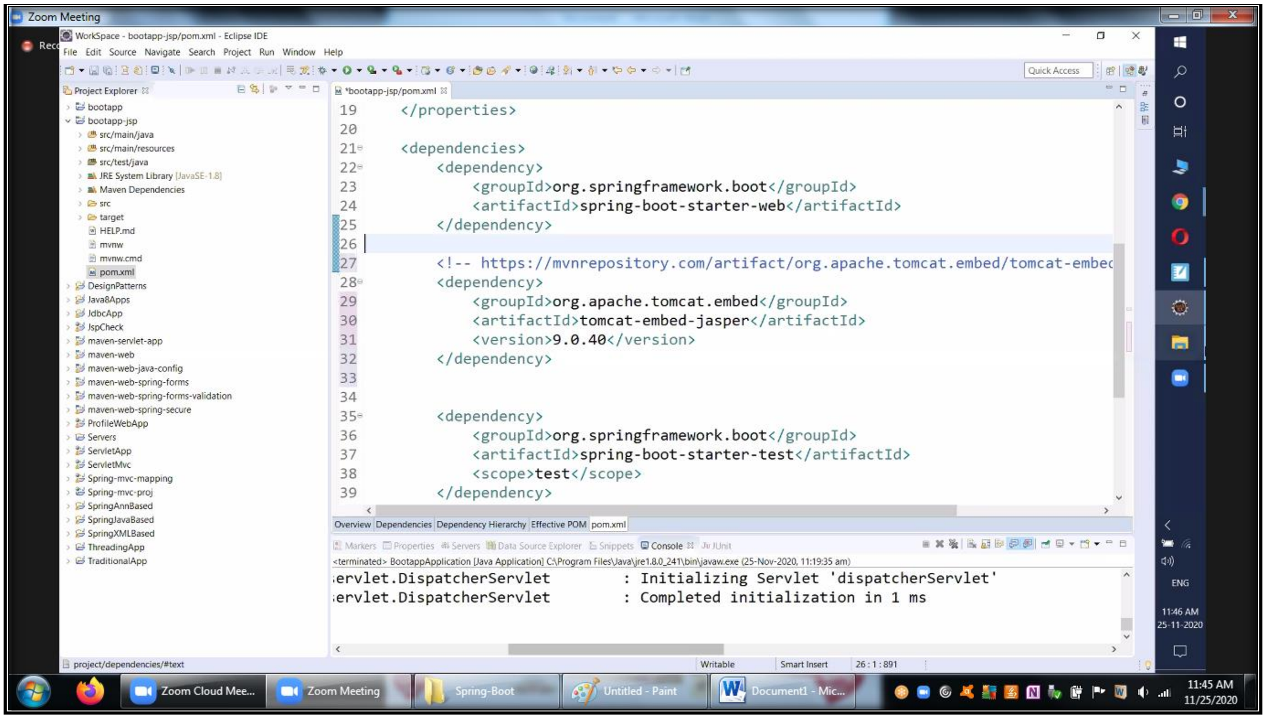
New Version	10.0.0-M10

Maven | Gradle | SBT | Ivy | Grape | Leiningen | Buildr

```
<!-- https://mvnrepository.com/artifact/org.apache.tomcat.embed/tomcat-embed-jasper -->
<dependency>
    <groupId>org.apache.tomcat.embed</groupId>
    <artifactId>tomcat-embed-jasper</artifactId>
    <version>9.0.40</version>
</dependency>
```

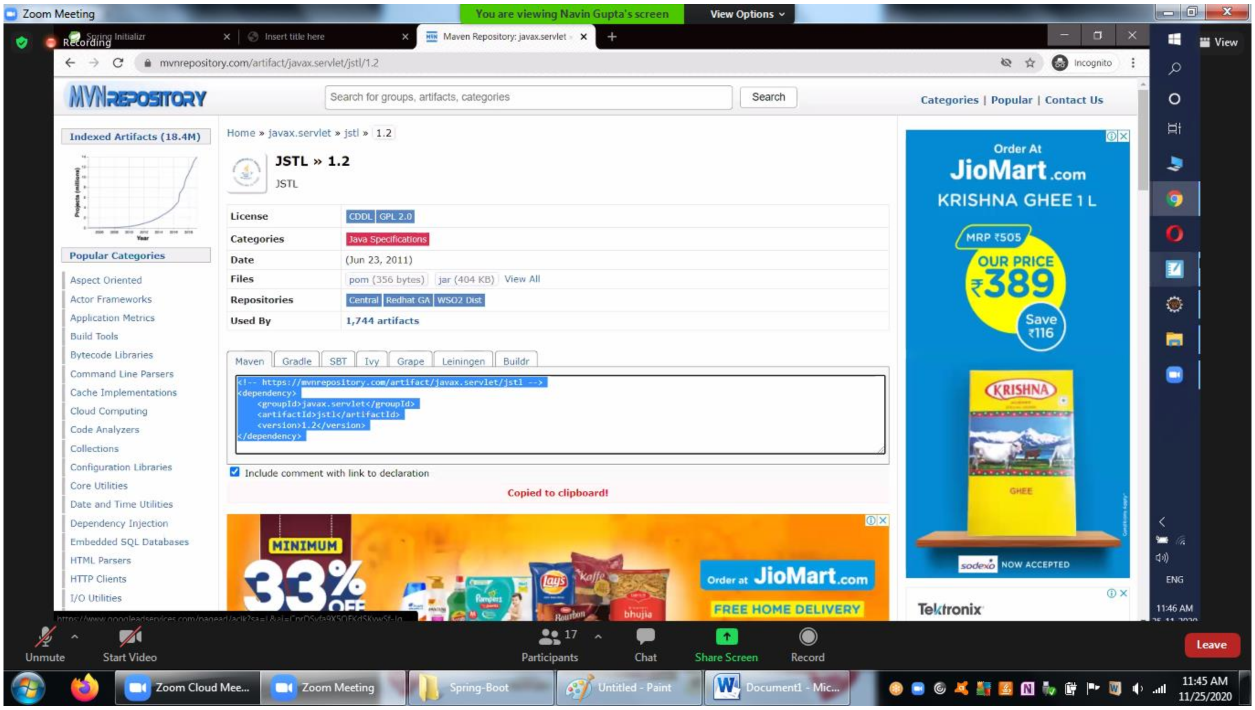
☑ Include comment with link to declaration

Remove Timeleaf and add the JSP and JSTL


WorkSpace - bootapp-jsp/pom.xml - Eclipse IDE
*bootapp-jsp/pom.xml
Overview Dependencies Dependency Hierarchy Effective POM pom.xml
Console

```
    </properties>

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

        <!-- https://mvnrepository.com/artifact/org.apache.tomcat.embed/tomcat-embec
        <dependency>
            <groupId>org.apache.tomcat.embed</groupId>
            <artifactId>tomcat-embed-jasper</artifactId>
            <version>9.0.40</version>
        </dependency>


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

```
<terminated> BootappApplication [Java Application] C:\Program Files\Java\jre1.8.0_241\bin\javaw.exe (25-Nov-2020, 11:19:35 am)
ervlet.DispatcherServlet        : Initializing Servlet 'dispatcherServlet'
ervlet.DispatcherServlet        : Completed initialization in 1 ms
```


Zoom Meeting
You are viewing Navin Gupta's screen
Maven Repository: javax.servlet
mvnrepository.com/artifact/javax.servlet/jstl/1.2
MVNREPOSITORY
Search
Categories | Popular | Contact Us

Home » javax.servlet » jstl » 1.2

JSTL » 1.2

JSTL

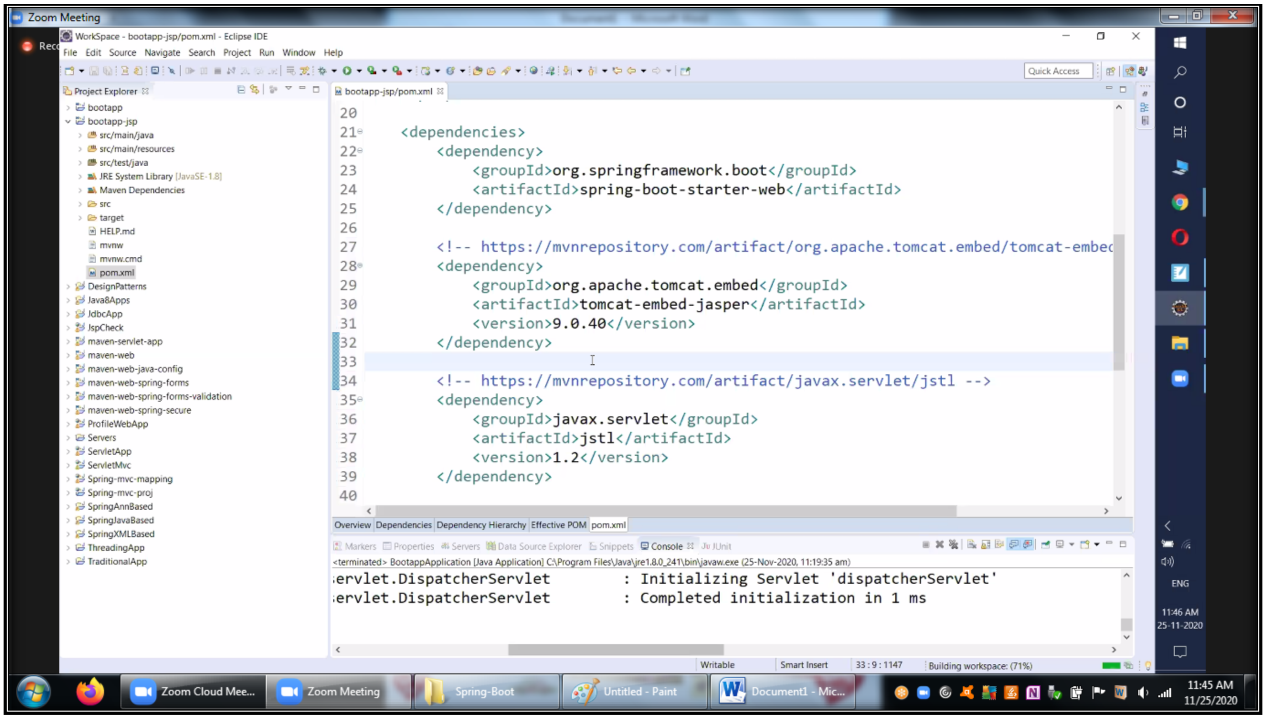
License	CDDL GPL 2.0
Categories	Java Specifications
Date	(Jun 23, 2011)
Files	pom (356 bytes) jar (404 KB) View All
Repositories	Central Redhat GA WSO2 Dist
Used By	1,744 artifacts

Maven Gradle SBT Ivy Grape Leiningen Buildr

```
<!-- https://mvnrepository.com/artifact/javax.servlet/jstl -->
<dependency>
    <groupId>javax.servlet</groupId>
    <artifactId>jstl</artifactId>
    <version>1.2</version>
</dependency>
```

☑ Include comment with link to declaration

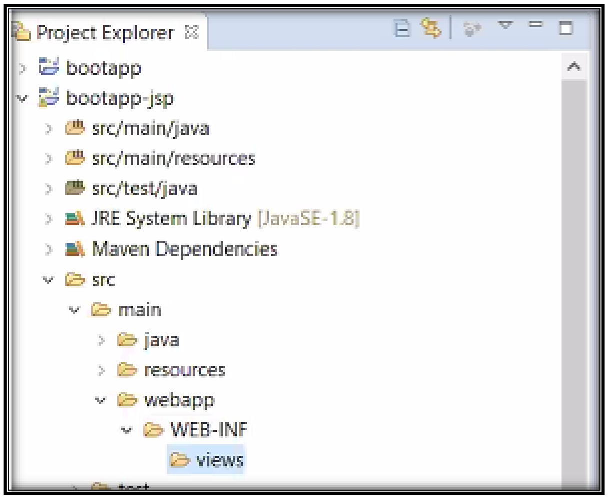
Copied to clipboard!

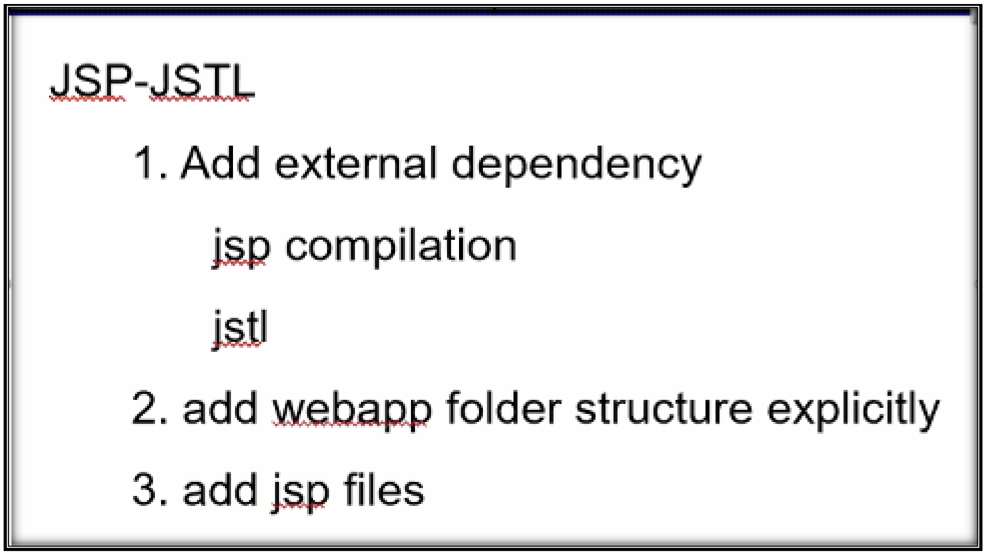


```

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

		<!-- https://mvnrepository.com/artifact/org.apache.tomcat.embed/tomcat-embec
		<dependency>
			<groupId>org.apache.tomcat.embed</groupId>
			<artifactId>tomcat-embed-jasper</artifactId>
			<version>9.0.40</version>
		</dependency>

		<!-- https://mvnrepository.com/artifact/javax.servlet/jstl -->
		<dependency>
			<groupId>javax.servlet</groupId>
			<artifactId>jstl</artifactId>
			<version>1.2</version>
		</dependency>

```

```
<terminated> BootappApplication [Java Application] C:\Program Files\Java\jre1.8.0_241\bin\javaw.exe (25-Nov-2020, 11:19:35 am)
ervlet.DispatcherServlet        : Initializing Servlet 'dispatcherServlet'
ervlet.DispatcherServlet        : Completed initialization in 1 ms
```

Add the webapp folder structure explicitly





JSP-JSTL

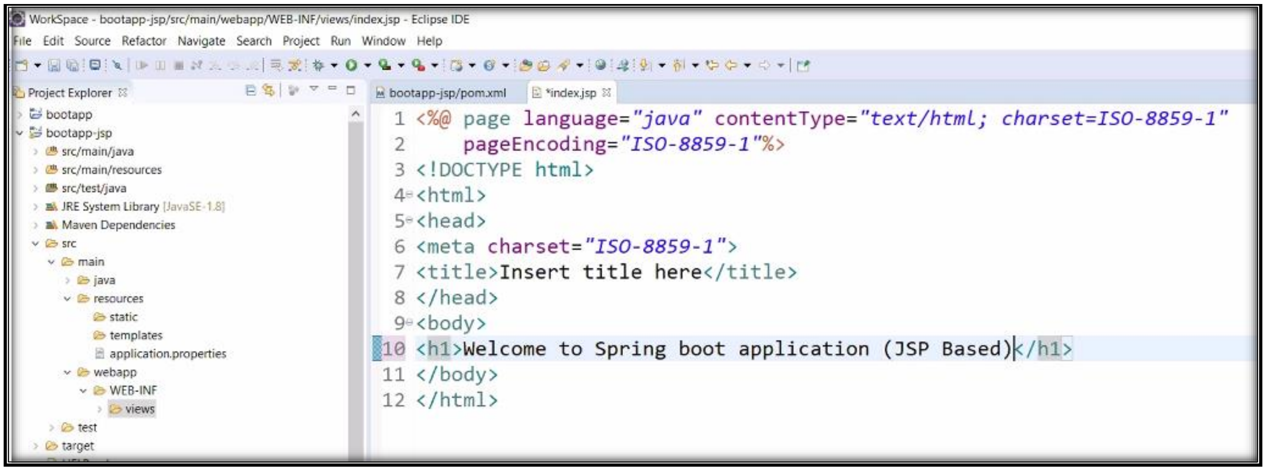
1. Add external dependency

   jsp compilation

   jstl

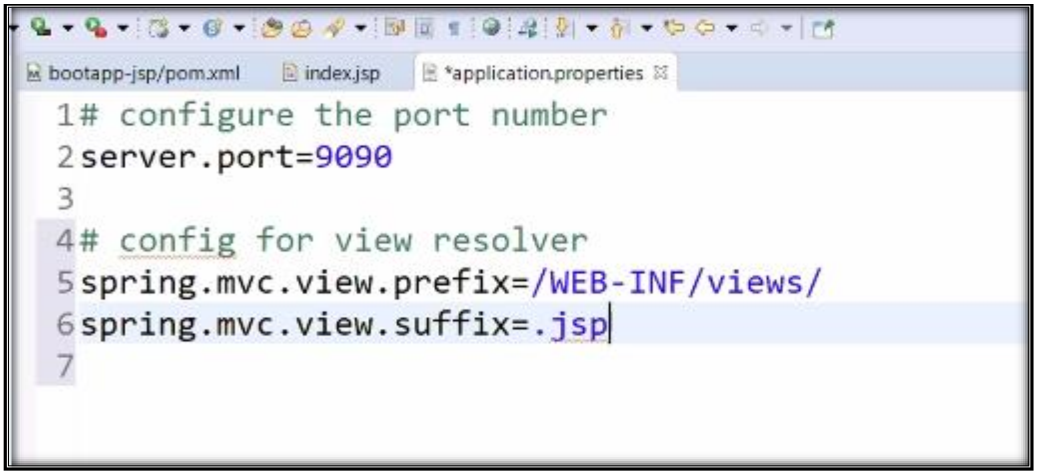
2. add webapp folder structure explicitly

3. add jsp files



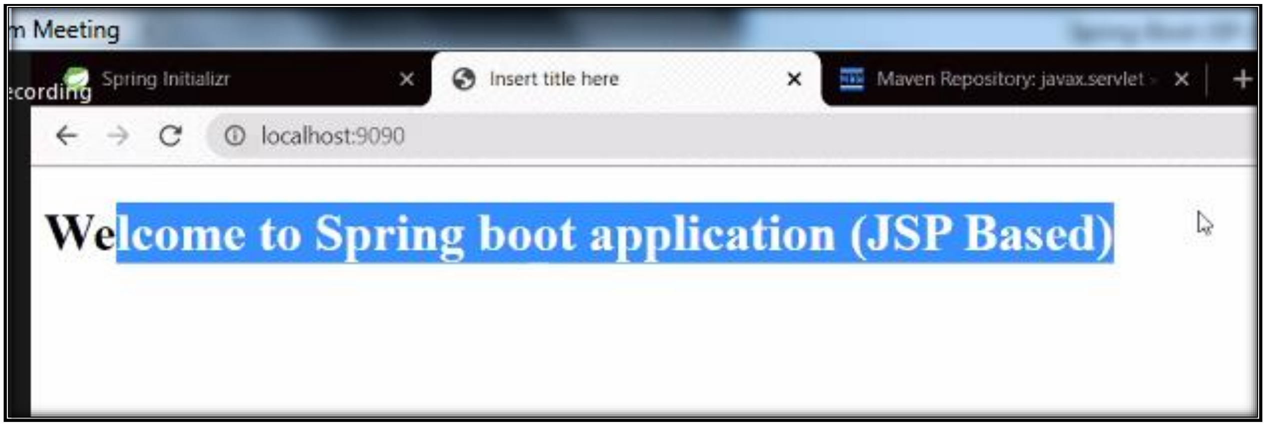
```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Insert title here</title>
</head>
<body>
<h1>Welcome to Spring boot application (JSP Based)</h1>
</body>
</html>
```

Add the Custom Config View Resolver, The same can be done using the properties file



```
# configure the port number
server.port=9090

# config for view resolver
spring.mvc.view.prefix=/WEB-INF/views/
spring.mvc.view.suffix=.jsp
```

Run as Java application and its success now



Welcome to Spring boot application (JSP Based)