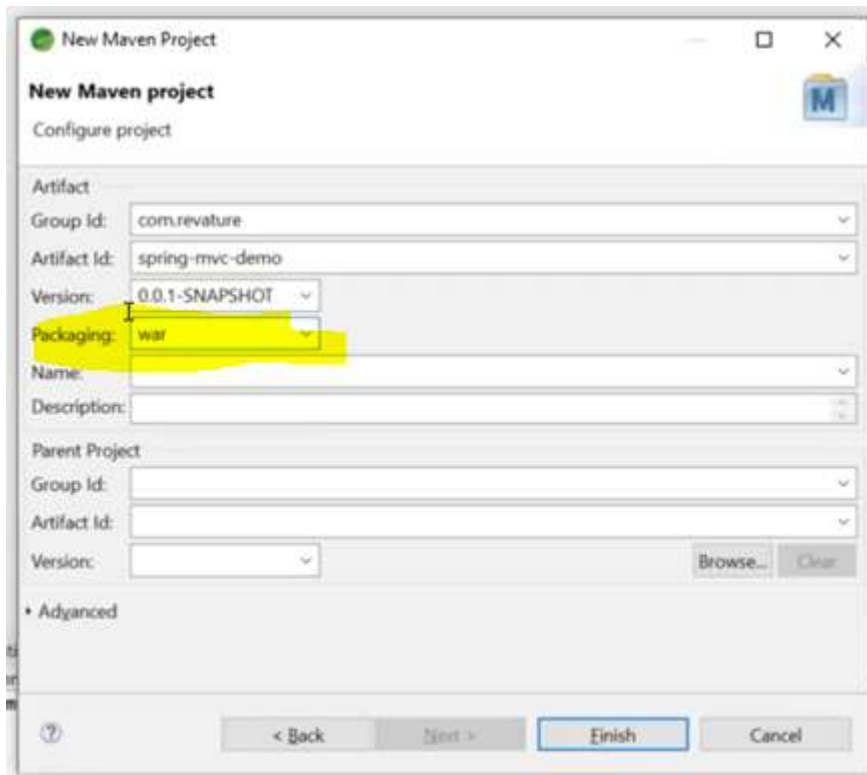


# Spring MVC

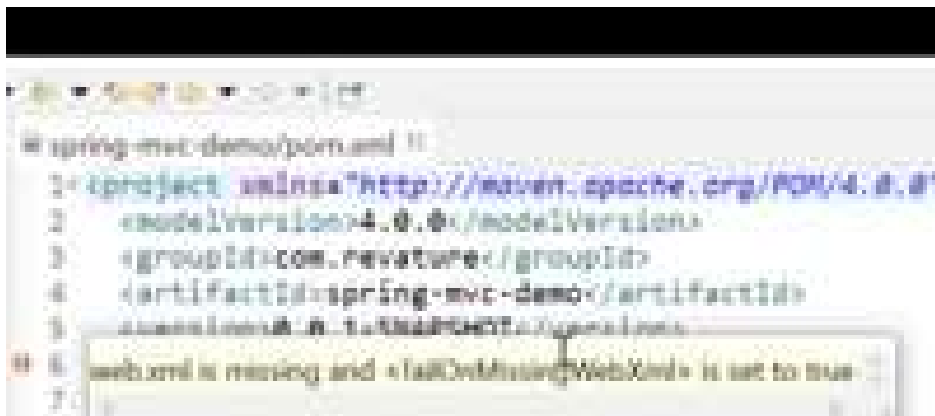
## Install / Use localhost Tomcat Server

spring-mvc-demo



Open pom.xml

Error: Web.xml is missing



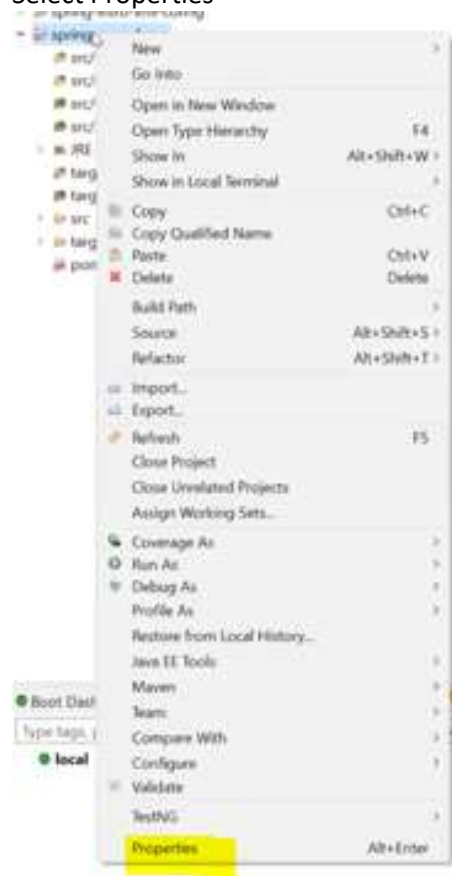
**BEFORE PROCEEDING MAKE SURE THE PROJECT IS ON Java 1.8 in the pom.xml and has been rebuilt.**

```
spring-mvc-demo/pom.xml
1 <project xmlns="http://maven.apache.org/POM/4.0.0" xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
2   <modelVersion>4.0.0</modelVersion>
3   <groupId>com.revature</groupId>
4   <artifactId>spring-mvc-demo</artifactId>
5   <version>0.0.1-SNAPSHOT</version>
6   <packaging>war</packaging>
7
8   <properties>
9     <maven.compiler.source>1.8</maven.compiler.source>
10    <maven.compiler.target>1.8</maven.compiler.target>
11  </properties>
12 </project>
```

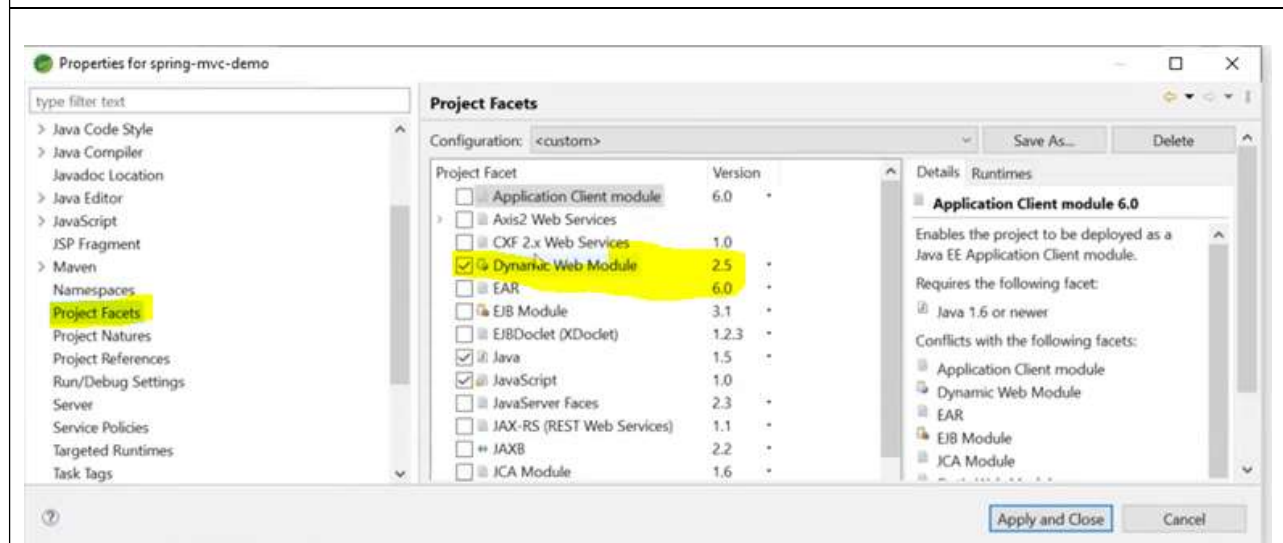
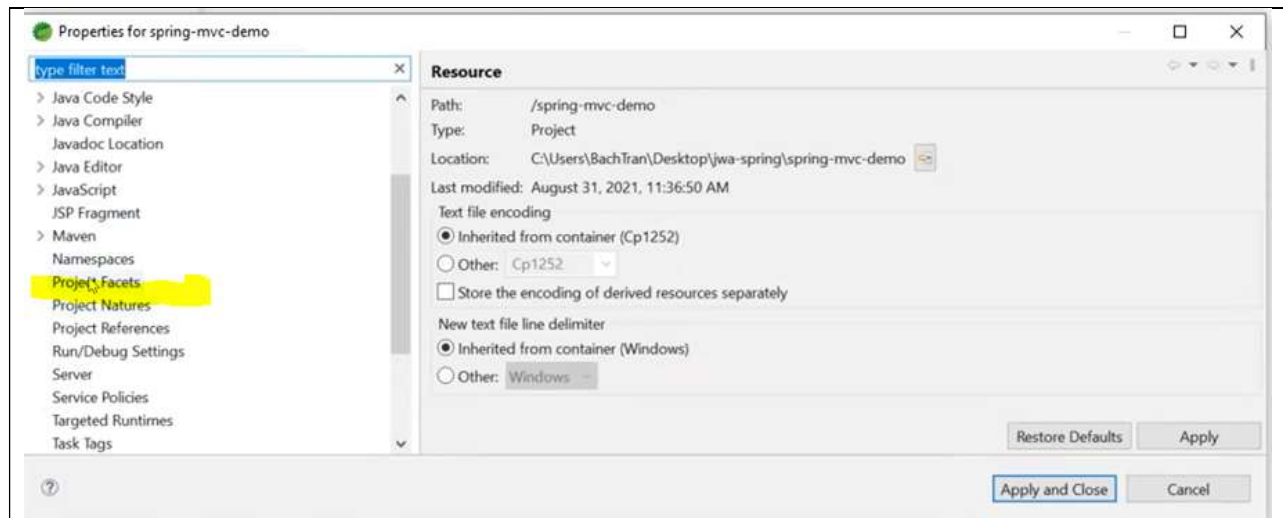
Need to generate Web.xml

Right click on project

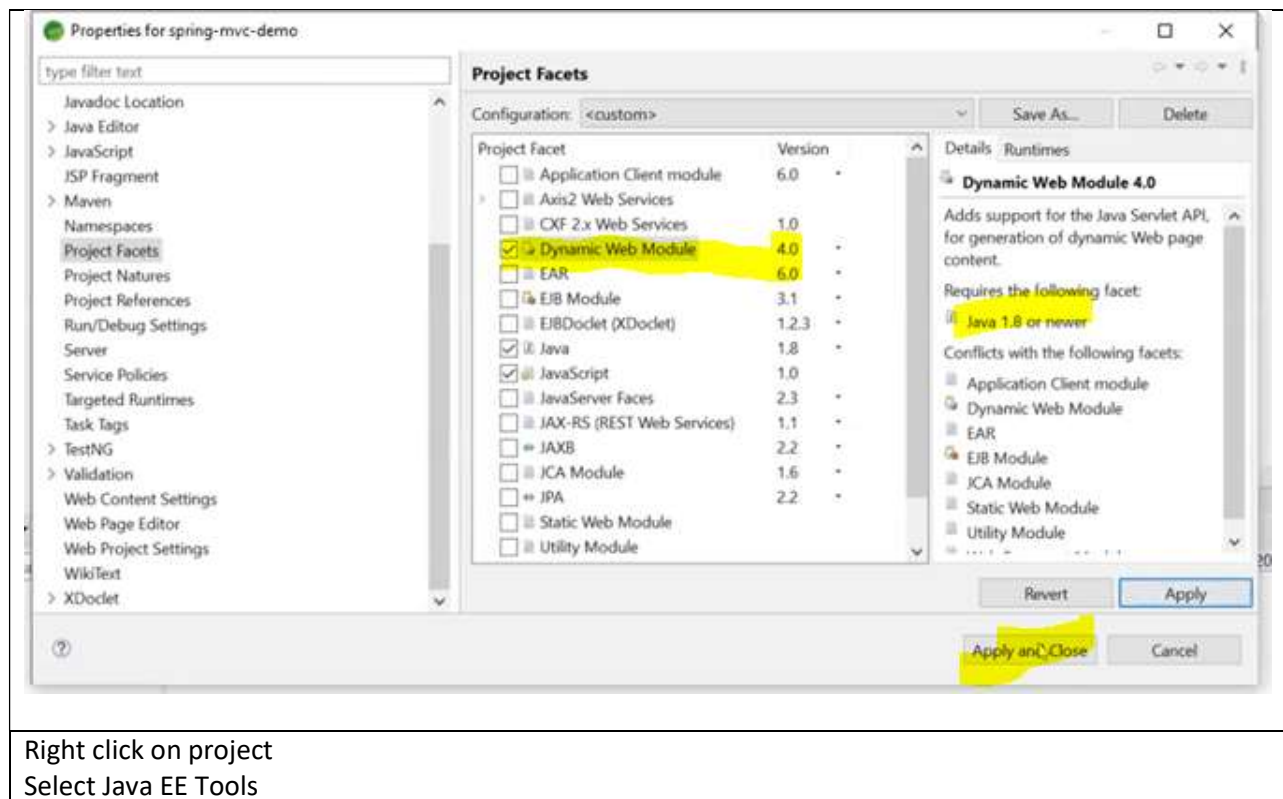
Select Properties



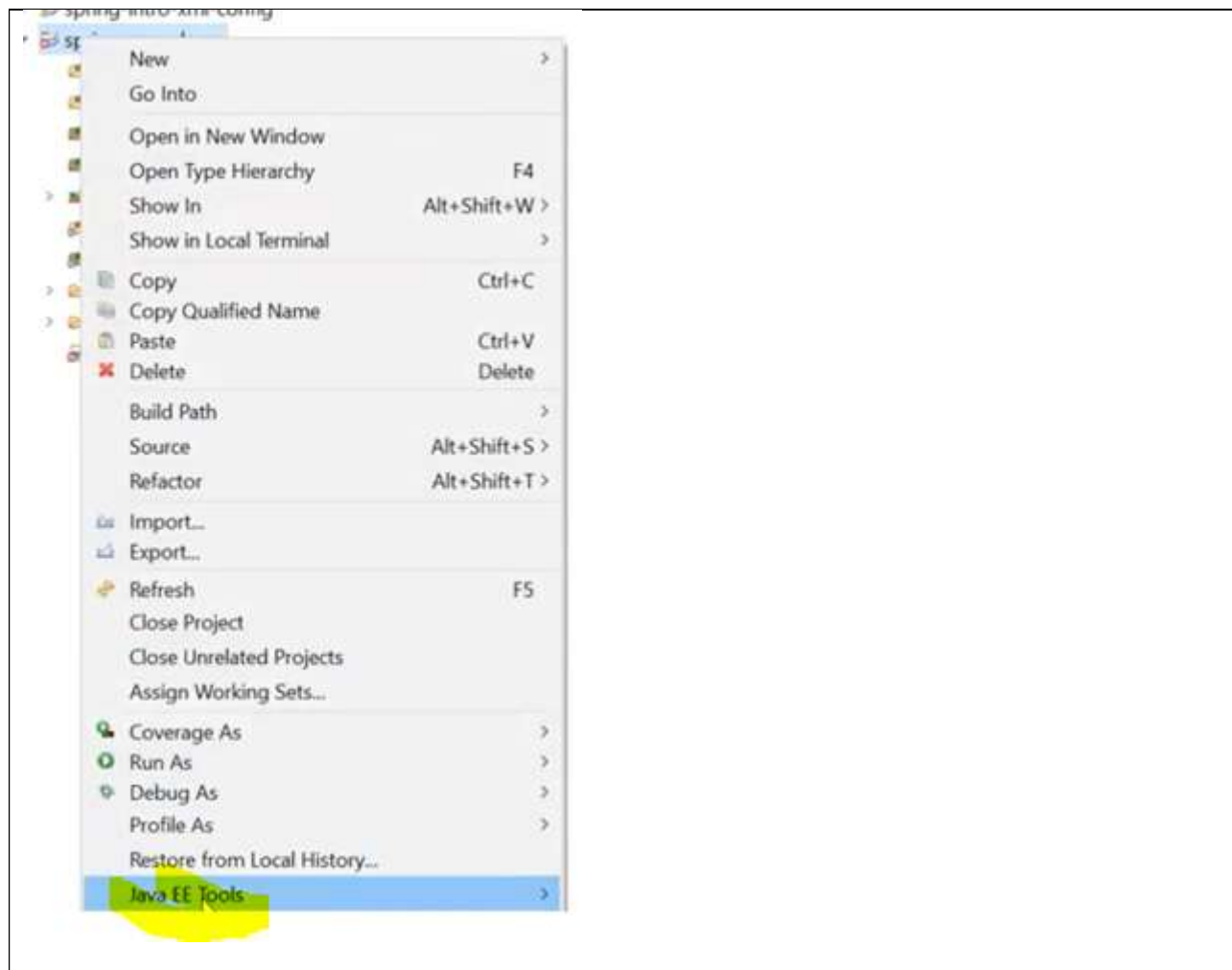
Select Project Facets



Change to new version:



Right click on project  
Select Java EE Tools



If Java EE Tools is not in the menu then:



Eclipse Marketplace

Eclipse Marketplace

Select solutions to install. Press Install Now to proceed with installation.  
Press the "more info" link to learn more about a solution.

Search

Recent

Popular

Favorites

Installed

Giving IoT an Edge


by Eclipse.org, EPL

[docker](#) [Containers](#) [Virtualization](#) [fileExtension](#) [Dockerfile](#)

★ 556

Installs: 143K (2,497 last month)

Uninstall



Eclipse Enterprise Java and Web Developer Tools 3.21

Enables Enterprise Java Bean, Java Enterprise Application, [Fragments](#), and Connector, Java Web Application, JavaServer Faces (JSF), Java Server Pages (JSP), Java... [more info](#)


by The Eclipse Foundation, EPL

[xml](#) [html](#) [CSS](#) [js](#) [jsp](#)

★ 1063

Installs: 559K (16,593 last month)

Change



Eclipse JST Server Adapters (Apache Tomcat, JOnAS, J2EE) Luna

This is quick way to install Eclipse JST Server Adapters and JST Server Adapters Extentions (Apache Tomcat, JOnAS, J2EE) This entry was created when Eclipse IDE... [more info](#)

by Nodeclipse/Enide, EPL


[wtp](#) [JST Server Adapters](#) [tomcat](#) [Apache Tomcat](#)

★ 180

Installs: 108K (1,078 last month)

Change

Marketplaces



?

< Back

Install Now >

Finish

Cancel

Search tab enter enterprise

Eclipse Marketplace

Eclipse Marketplace

Select solutions to install. Press Install Now to proceed with installation.  
Press the "more info" link to learn more about a solution.

SearchRecentPopularFavoritesInstalledGiving IoT an Edge

Find:enterpriseAll MarketsAll CategoriesGo

Eclipse Enterprise Java and Web Developer Tools 3.21

Enables Enterprise Java Bean, Java Enterprise Application, Fragments, and Connector, Java Web Application, JavaServer Faces (JSF), Java Server Pages (JSP), Java... more info

by The Eclipse Foundation, EPL  
xmlhtmlCSSjsjsp

★ 1063Installs: 559K (16,593 last month)Installed

PexTools

Open Shell & Open File/Folder in Explorer & Copy File Path to Clipboard with no hassles!! After this plugin is installed,the right click menu on any item... more info

by Abhinandan M Kulkarni, GPL  
developer tools Eclipse Plugin

★ 13Installs: 2.20K (3 last month)Install

Export plugin to the requests console for 1C:EDT 1.0.0

Marketplaces

?

< Back

Install Now >

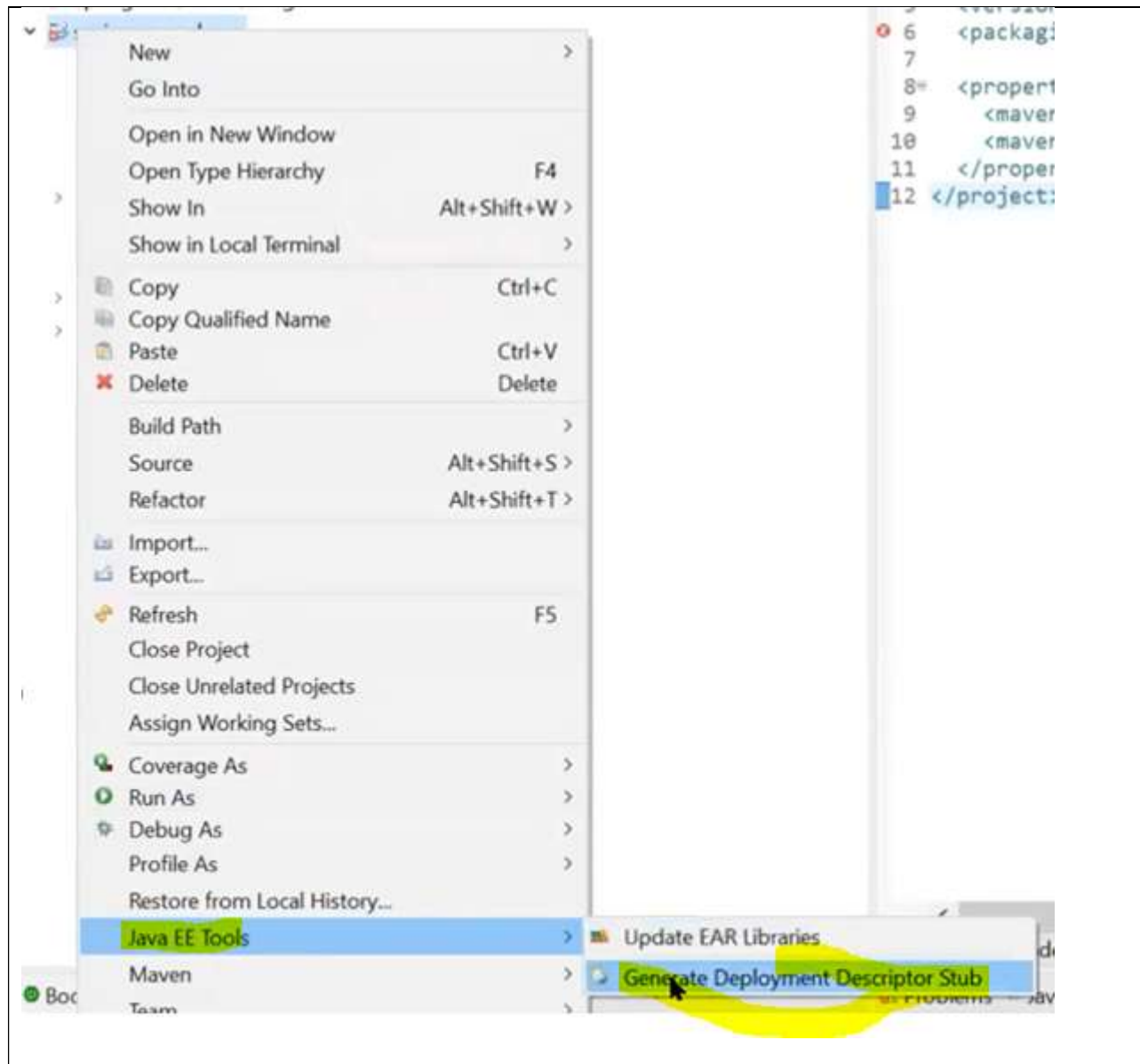
Finish

Cancel

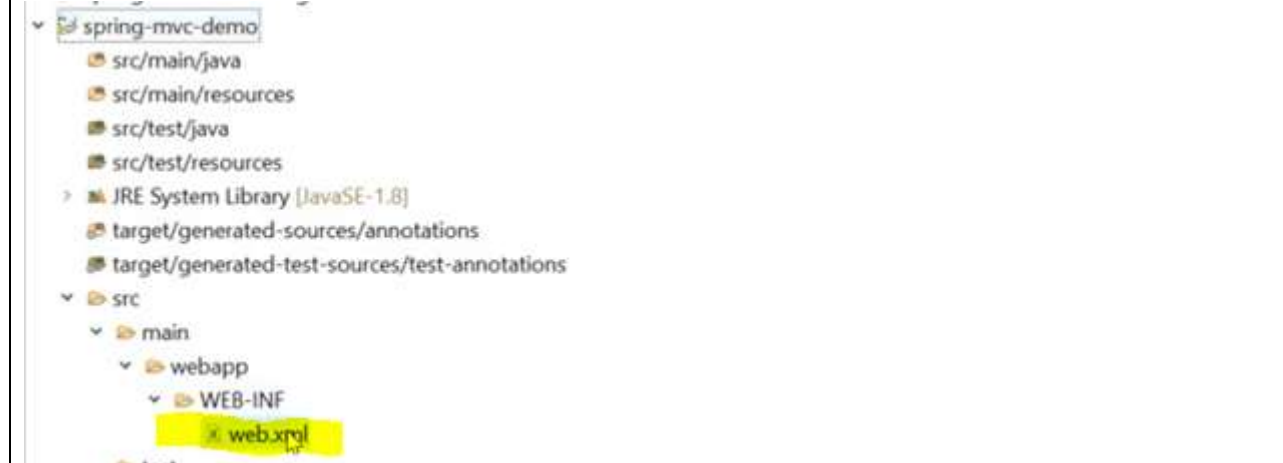
Now right click project

Select Java EE Tools

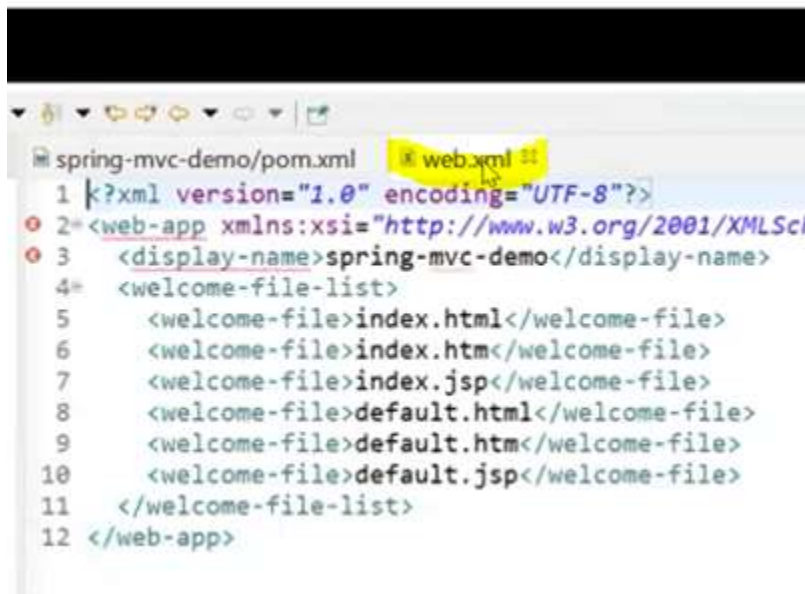
Generate Deployment Descriptor Stub



Generates web.xml in following location







```
spring-mvc-demo/pom.xml  web.xml
1 <?xml version="1.0" encoding="UTF-8"?>
2 <web-app xmlns:xsi="http://www.w3.org/2001/XMLSchema"
3   <display-name>spring-mvc-demo</display-name>
4   <welcome-file-list>
5     <welcome-file>index.html</welcome-file>
6     <welcome-file>index.htm</welcome-file>
7     <welcome-file>index.jsp</welcome-file>
8     <welcome-file>default.html</welcome-file>
9     <welcome-file>default.htm</welcome-file>
10    <welcome-file>default.jsp</welcome-file>
11  </welcome-file-list>
12 </web-app>
```

Update outdated link:

```
<web-app xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns="http://xmlns.jcp.org/xml/ns/javaee"
xsi:schemaLocation="http://xmlns.jcp.org/xml/ns/javaee
http://xmlns.jcp.org/xml/ns/javaee/web-app_4_0.xsd" version="4.0">
```



```
spring-mvc-demo/pom.xml  *web.xml
1 encoding="UTF-8"?>
2 http://www.w3.org/2001/XMLSchema-instance" xmlns="http://www.oracle.com/webfolder/technetwork/jsc/xml/n
3 ng-mvc-demo</display-name>
4 >
```

web.xml is for the Tomcat server configuration.

Add DispatcherServlet to web.xml

```
<!-- This is where we configure our DispatcherServlet (which comes from Spring Web) -->
<!-- The DispatcherServlet is the sole Servlet that receives HTTP requests through our Tomcat server
-->
<!-- It then routes the HTTP requests to the appropriate controller -->
<servlet>
  <servlet-name>DispatcherServlet</servlet-name>
  <servlet-class>org.springframework.web.servlet.DispatcherServlet</servlet-class>

  <init-param>
    <param-name>contextConfigLocation</param-name>
    <param-value>/WEB-INF/applicationContext.xml</param-value>
  </init-param>
  <load-on-startup>1</load-on-startup>
</servlet>

<servlet-mapping>
  <servlet-name>DispatcherServlet</servlet-name>
  <url-pattern>/</url-pattern>
```

```

</servlet-mapping>
28< <servlet-mapping>
29    <servlet-name>DispatcherServlet</servlet-name>
30    <url-pattern>/</url-pattern>
31  </servlet-mapping>
32

```

All http request goes to the DispatcherServlet which in turn sends to the controller(s) endpoints.

"Front Controller Design Pattern"

### applicationContext.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"
       xmlns:context="http://www.springframework.org/schema/context"
       xmlns:mvc="http://www.springframework.org/schema/mvc"
       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.revature"></context:component-
scan>

    <mvc:annotation-driven></mvc:annotation-driven>

</beans>

```



```

web.xml  applicationContext.xml
1 <?xml version="1.0" encoding="UTF-8"?>
2 <beans xmlns="http://www.springframework.org/schema/beans"
3     xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
4     xmlns:context="http://www.springframework.org/schema/context"
5     xmlns:mvc="http://www.springframework.org/schema/mvc"
6     xsi:schemaLocation="http://www.springframework.org/schema/beans
7     http://www.springframework.org/schema/beans/spring-beans.xsd
8     http://www.springframework.org/schema/context
9     http://www.springframework.org/schema/context/spring-context.xsd
10    http://www.springframework.org/schema/mvc
11    http://www.springframework.org/schema/mvc/spring-mvc.xsd">
12
13    <context:component-scan base-package="com.revature"></context:component-scan>
14
15 </beans>

```

mcv:annotation allows for the @Component, @Service and @RestController tags.

```

web.xml applicationContext.xml
1 <?xml version="1.0" encoding="UTF-8"?>
2 <beans xmlns="http://www.springframework.org/schema/beans"
3     xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
4     xmlns:context="http://www.springframework.org/schema/context"
5     xmlns:mvc="http://www.springframework.org/schema/mvc"
6     xsi:schemaLocation="http://www.springframework.org/schema/beans
7     http://www.springframework.org/schema/beans/spring-beans.xsd
8     http://www.springframework.org/schema/context
9     http://www.springframework.org/schema/context/spring-context.xsd
10    http://www.springframework.org/schema/mvc
11    http://www.springframework.org/schema/mvc/spring-mvc.xsd">
12
13     <context:component-scan base-package="com.revature"></context:component-scan>
14     <mvc:annotation-driven></mvc:annotation-driven>
15
16 </beans>

```

#### Add other project dependencies to 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.revature</groupId>
  <artifactId>spring-mvc-demo</artifactId>
  <version>0.0.1-SNAPSHOT</version>
  <packaging>war</packaging>

  <properties>
    <maven.compiler.source>1.8</maven.compiler.source>
    <maven.compiler.target>1.8</maven.compiler.target>
  </properties>

  <dependencies>
    <!-- https://mvnrepository.com/artifact/javax.servlet/javax.servlet-api -->
    <dependency>
      <groupId>javax.servlet</groupId>
      <artifactId>javax.servlet-api</artifactId>
      <version>4.0.1</version>
      <scope>provided</scope>
    </dependency>
    <!-- https://mvnrepository.com/artifact/com.fasterxml.jackson.core/jackson-databind -->
    <dependency>
      <groupId>com.fasterxml.jackson.core</groupId>
      <artifactId>jackson-databind</artifactId>
      <version>2.12.5</version>
    </dependency>
    <!-- https://mvnrepository.com/artifact/org.springframework/spring-webmvc -->
    <dependency>
      <groupId>org.springframework</groupId>
      <artifactId>spring-webmvc</artifactId>
      <version>5.3.9</version>
    </dependency>
    <!-- https://mvnrepository.com/artifact/org.projectlombok/lombok -->
    <dependency>
      <groupId>org.projectlombok</groupId>
      <artifactId>lombok</artifactId>
      <version>1.18.20</version>
      <scope>provided</scope>
    </dependency>
  </dependencies>

```

</project>

Tomcat server

<https://tomcat.apache.org/download-90.cgi>

Download to folder and unzip, do not have to install

9.0.52

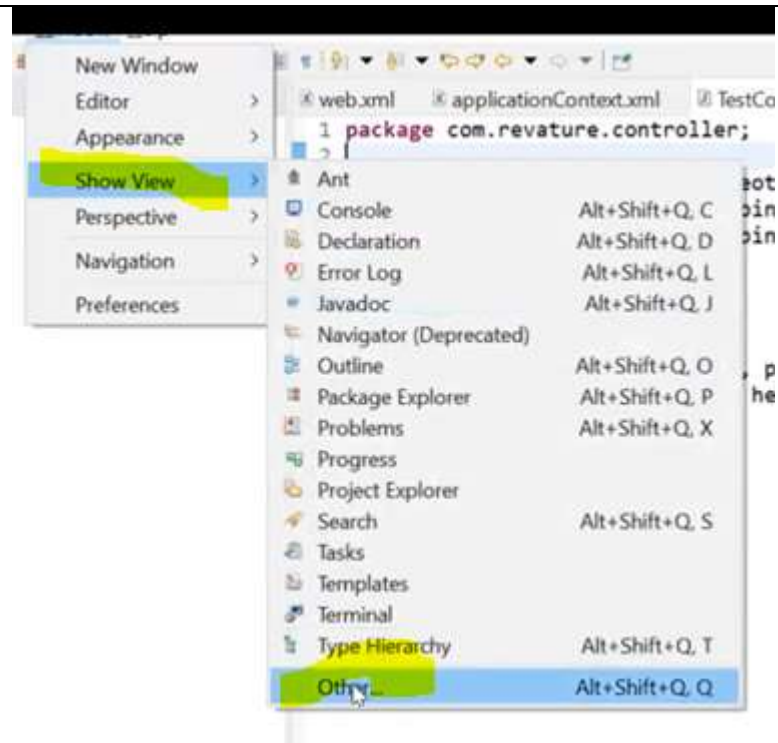
Please see the [README](#) file for packaging information. It explains what every c

#### Binary Distributions

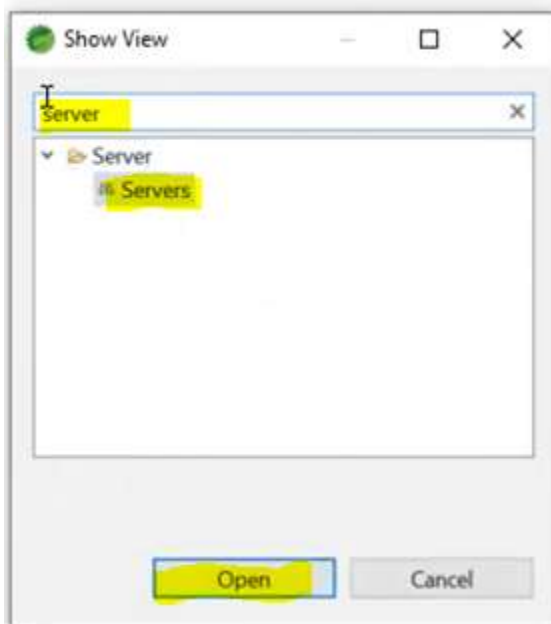
- Core:
  - [zip](#) ([pgp](#), [sha512](#))
  - [tar.gz](#) ([pgp](#), [sha512](#))
  - [32-bit Windows zip](#) ([pgp](#), [sha512](#))
  - [64-bit Windows zip](#) ([pgp](#), [sha512](#))
  - [32-bit/64-bit Windows Service Installer](#) ([pgp](#), [sha512](#))
- Full documentation:
  - [tar.gz](#) ([pgp](#), [sha512](#))
- Deployer:
  - [zip](#) ([pgp](#), [sha512](#))
  - [tar.gz](#) ([pgp](#), [sha512](#))
- Embedded:
  - [tar.gz](#) ([pgp](#), [sha512](#))
  - [zip](#) ([pgp](#), [sha512](#))

Configure Tomcat server in Spring Tools IDE

Window → Show View → Other



Search for Server



web.xml applicationContext.xml TestController.java spring-mvc-demo/pom.xml

```
1 package com.revature.controller;
2
3 import org.springframework.stereotype.Controller;
4 import org.springframework.web.bind.annotation.GetMapping;
5 import org.springframework.web.bind.annotation.ResponseBody;
6
7 @Controller
8 public class TestController {
9
10     @GetMapping(path = "/hello", produces = "application/json" )
11     public @ResponseBody String hello() {
12         return "Hello world!";
13     }
14 }
15 }
16
```

Problems Javadoc Declaration Console Servers

No servers are available. [Click this link to create a new server](#)

Click link to add a new server

```
8 public class TestController {  
9  
10     @GetMapping(path =  
11     public @ResponseBody  
12     return "Hello";  
13 }  
14  
15 }  
16
```

New Server

Define a New Server

Choose the type of server to create

Select the server type:

type filter text

HTTP Draculian

A server to preview static Web modules.

Server's host name: localhost

Server name: HTTP Preview at localhost

Help < Back Next > Finish Cancel

Problems Javadoc Declaration Console Servers

to servers are available. Click this link to create a new server...



New Server

Define a New Server

Choose the type of server to create

Select the server type:

type filter text

Apache

Tomcat v3.2 Server

Tomcat v4.0 Server

Tomcat v4.1 Server

Tomcat v5.0 Server

Tomcat v5.5 Server

Tomcat v6.0 Server

Tomcat v7.0 Server

Tomcat v8.0 Server

Tomcat v8.5 Server

Tomcat v9.0 Server

Tomcat v10.0 Server

Basic

Publishes and runs J2EE and Java EE Web projects and server configurations to a local Tomcat server.

Server's host name:

localhost

Server name:

Tomcat v9.0 Server at localhost

?

< Back

Next >

Finish

Cancel



New Server

Tomcat Server

Specify the installation directory

Name:

Apache Tomcat v9.0

Tomcat installation directory:

Browse...

apache-tomcat-9.0.43

Download and Install...

JRE:

Workbench default JRE

Installed JREs...

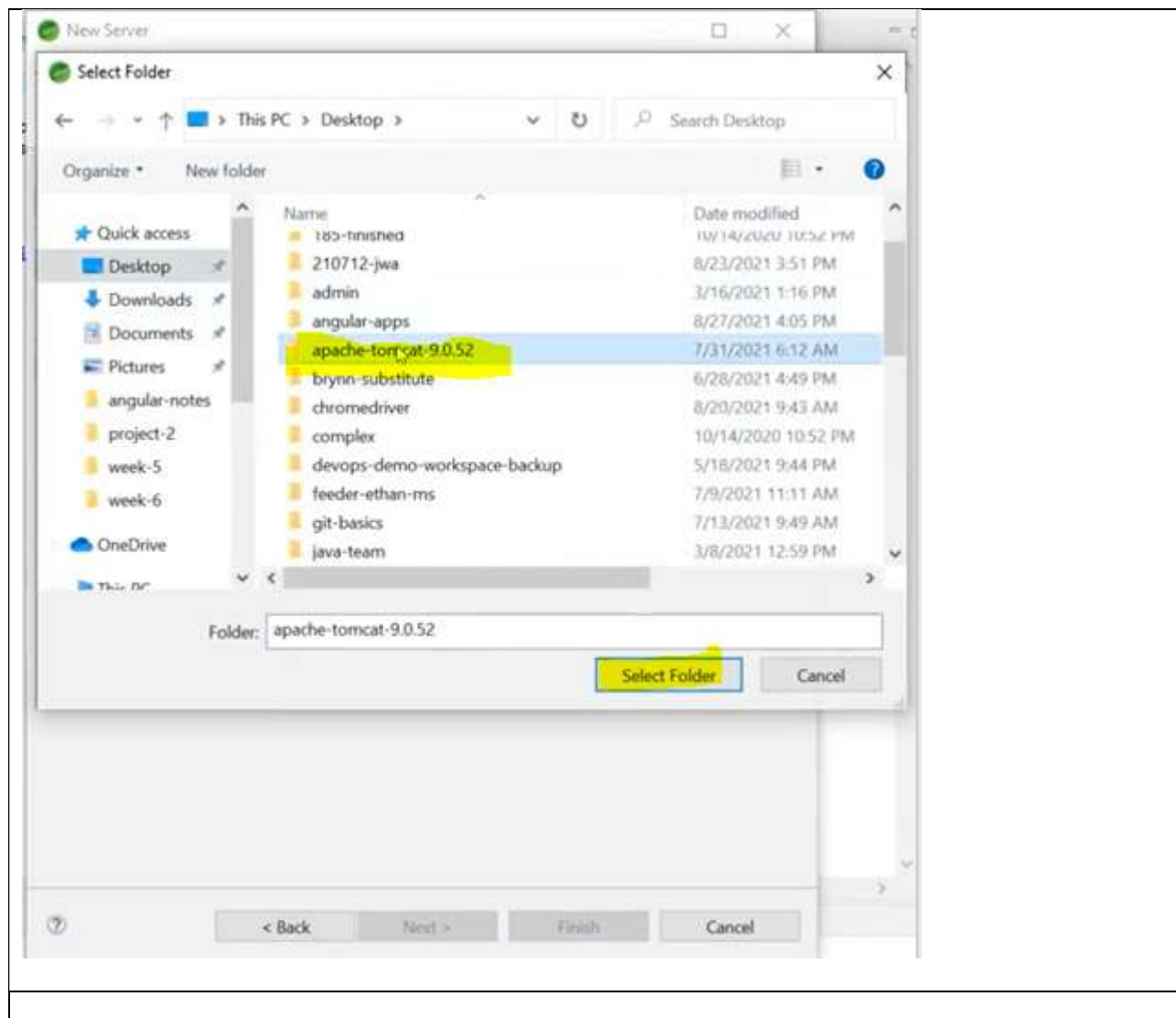
?

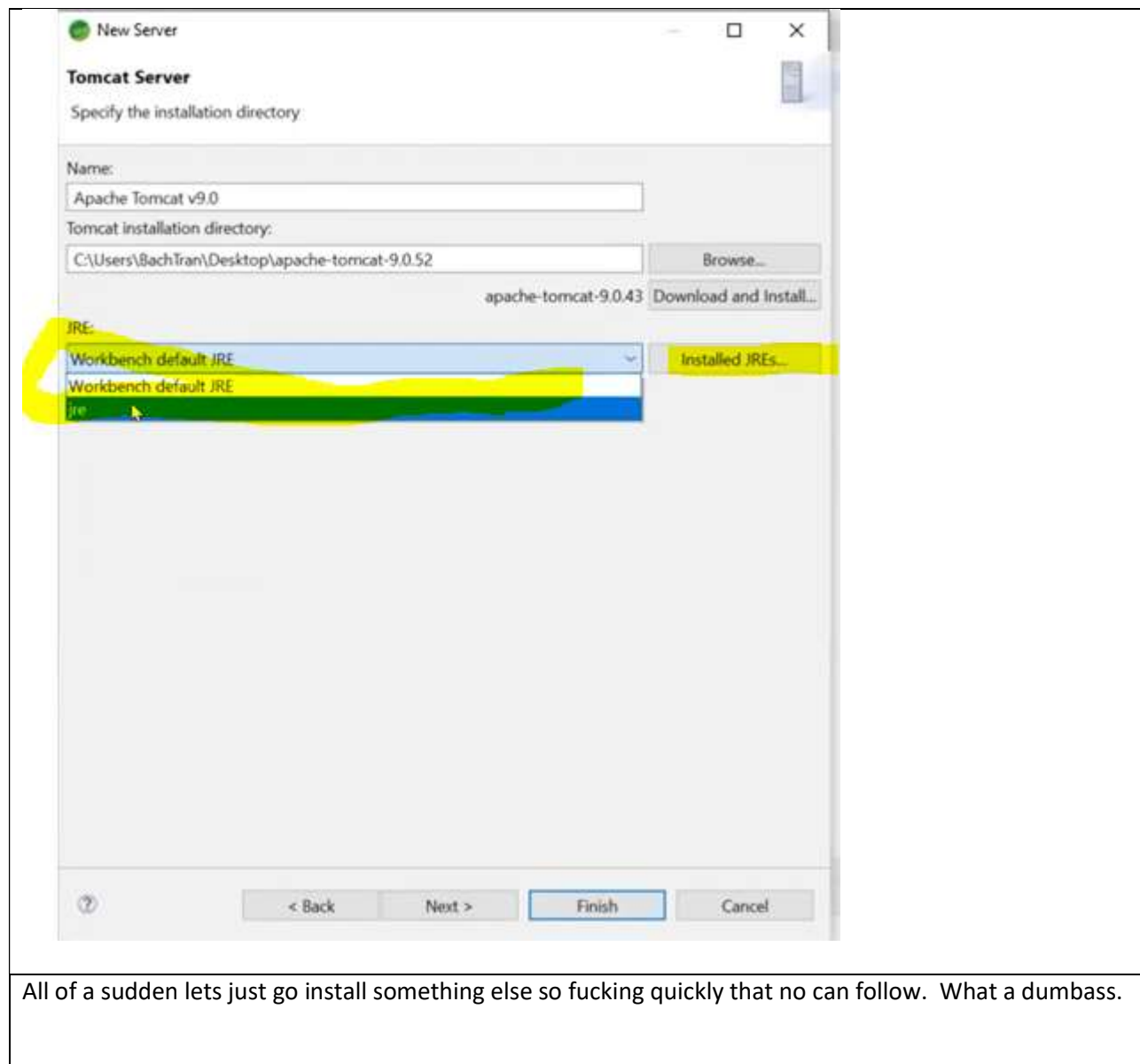
< Back

Next >

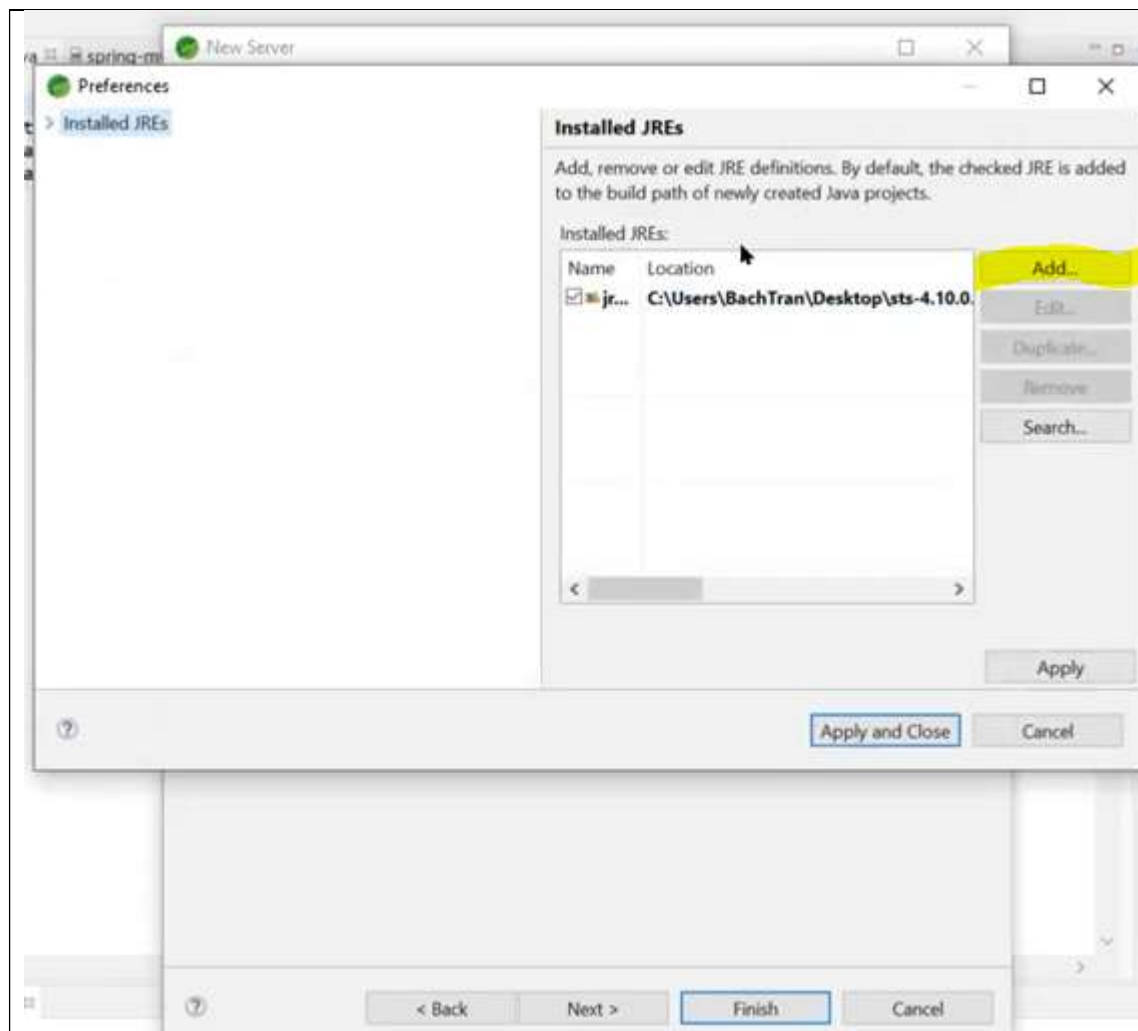
Finish

Cancel

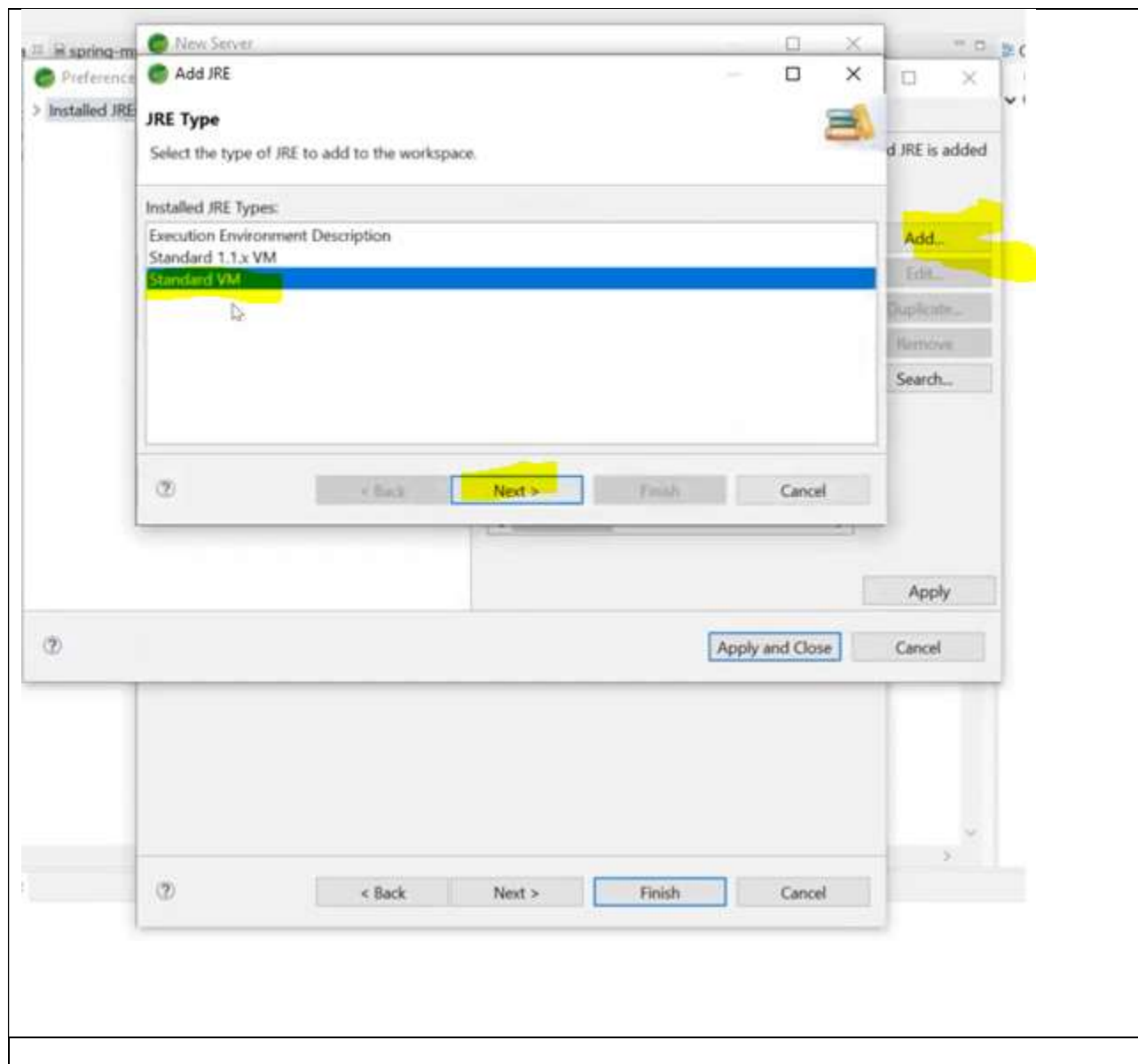


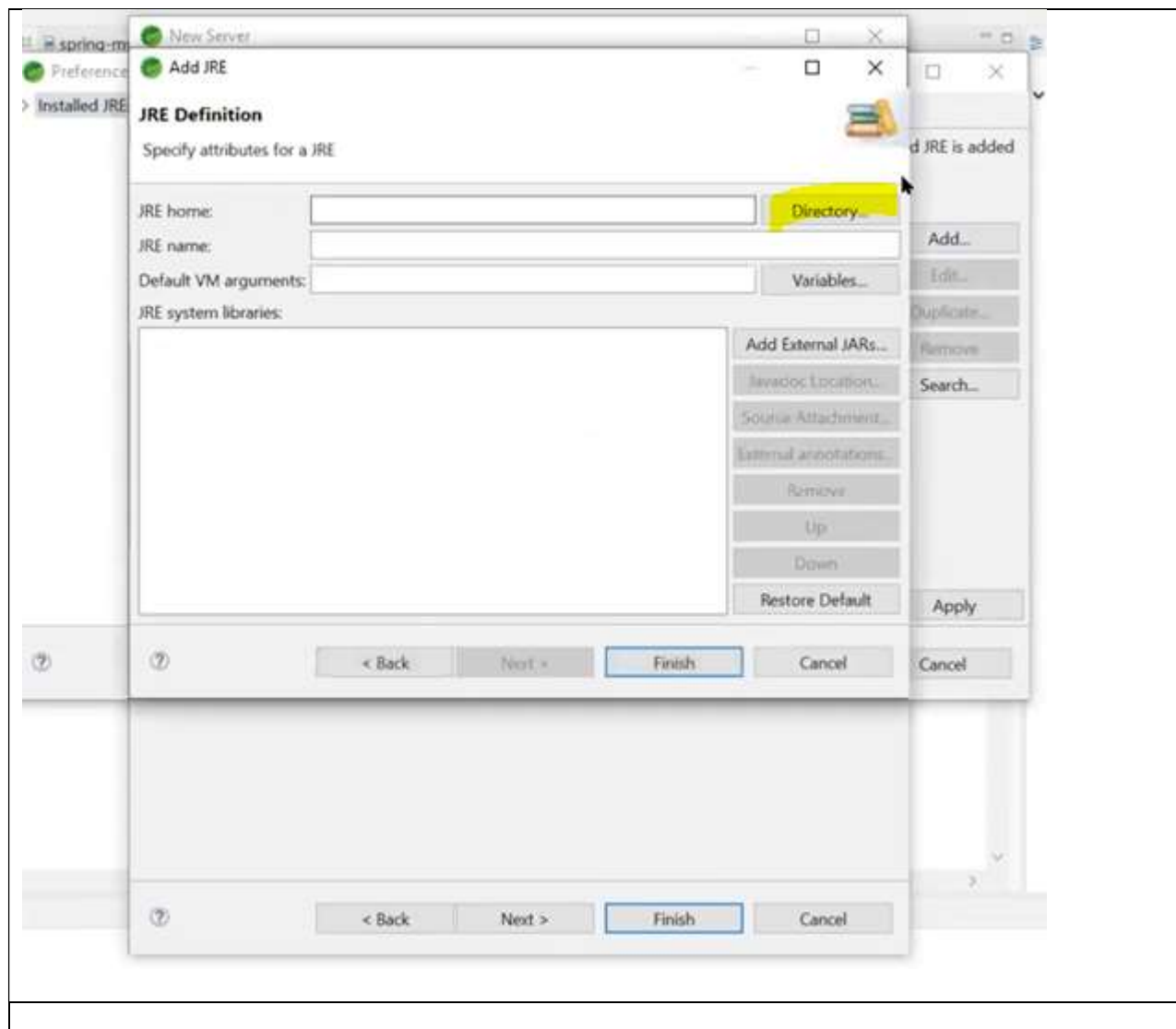


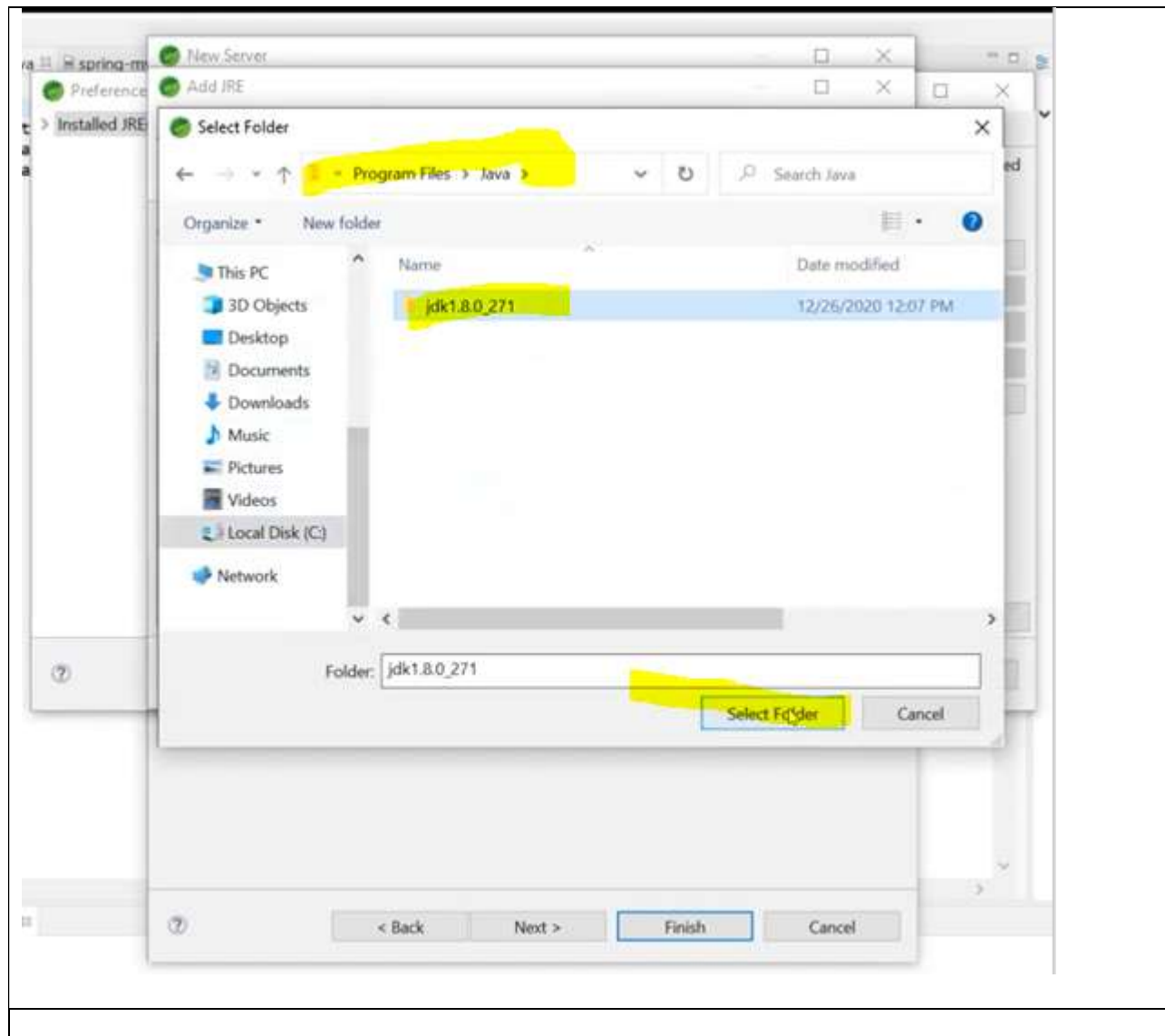
All of a sudden lets just go install something else so fucking quickly that no can follow. What a dumbass.

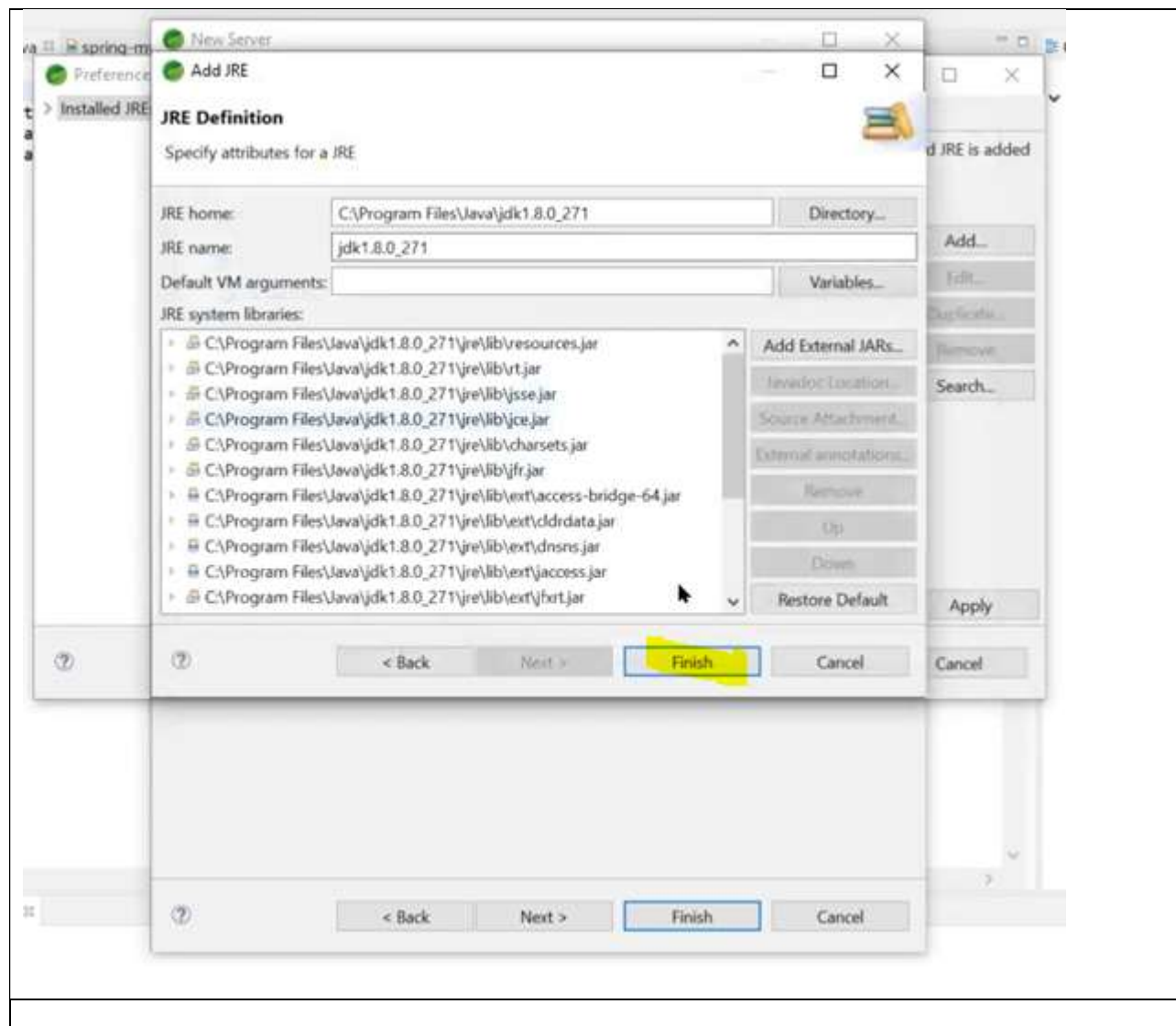


Add the Java JRE from your machine

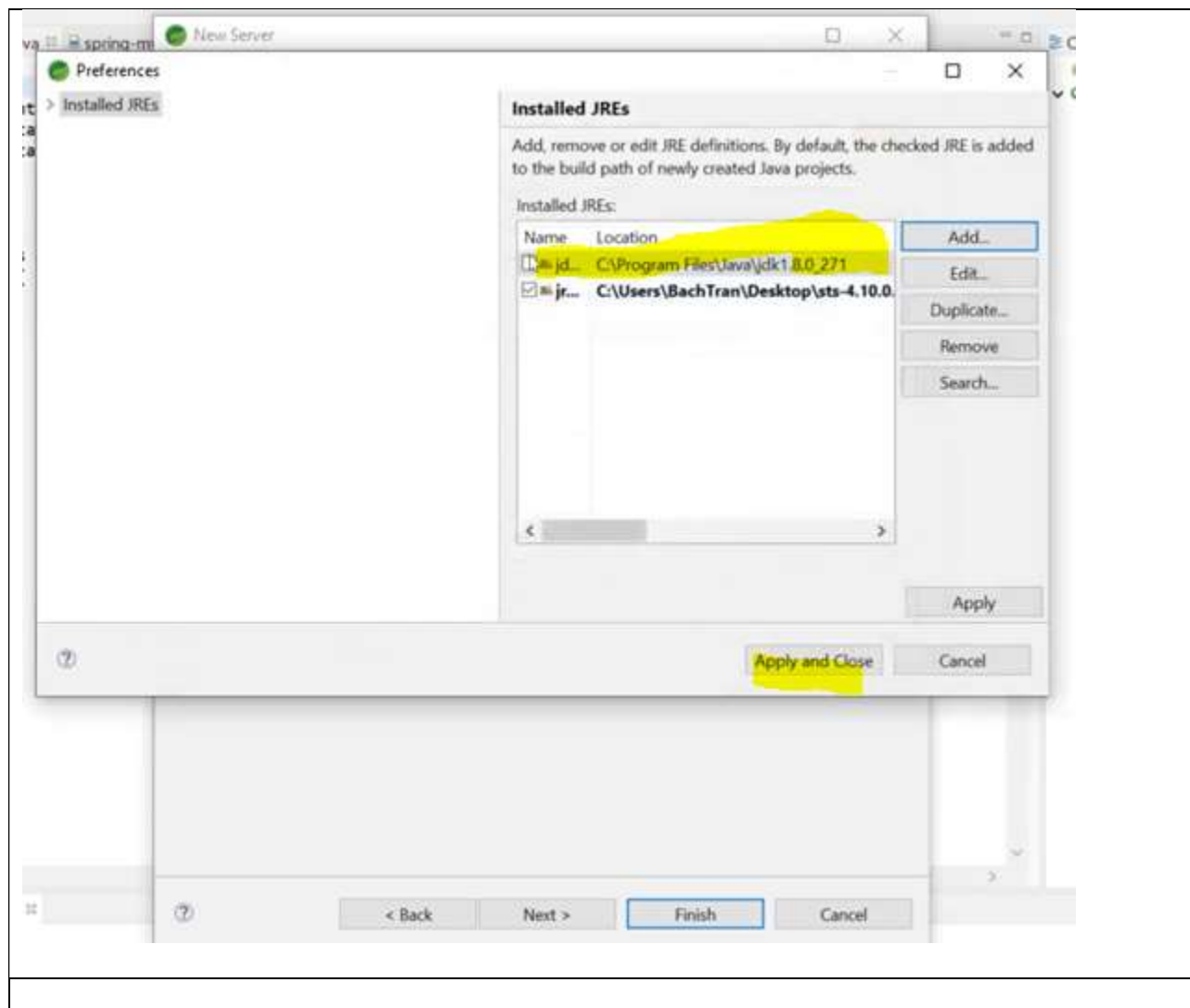




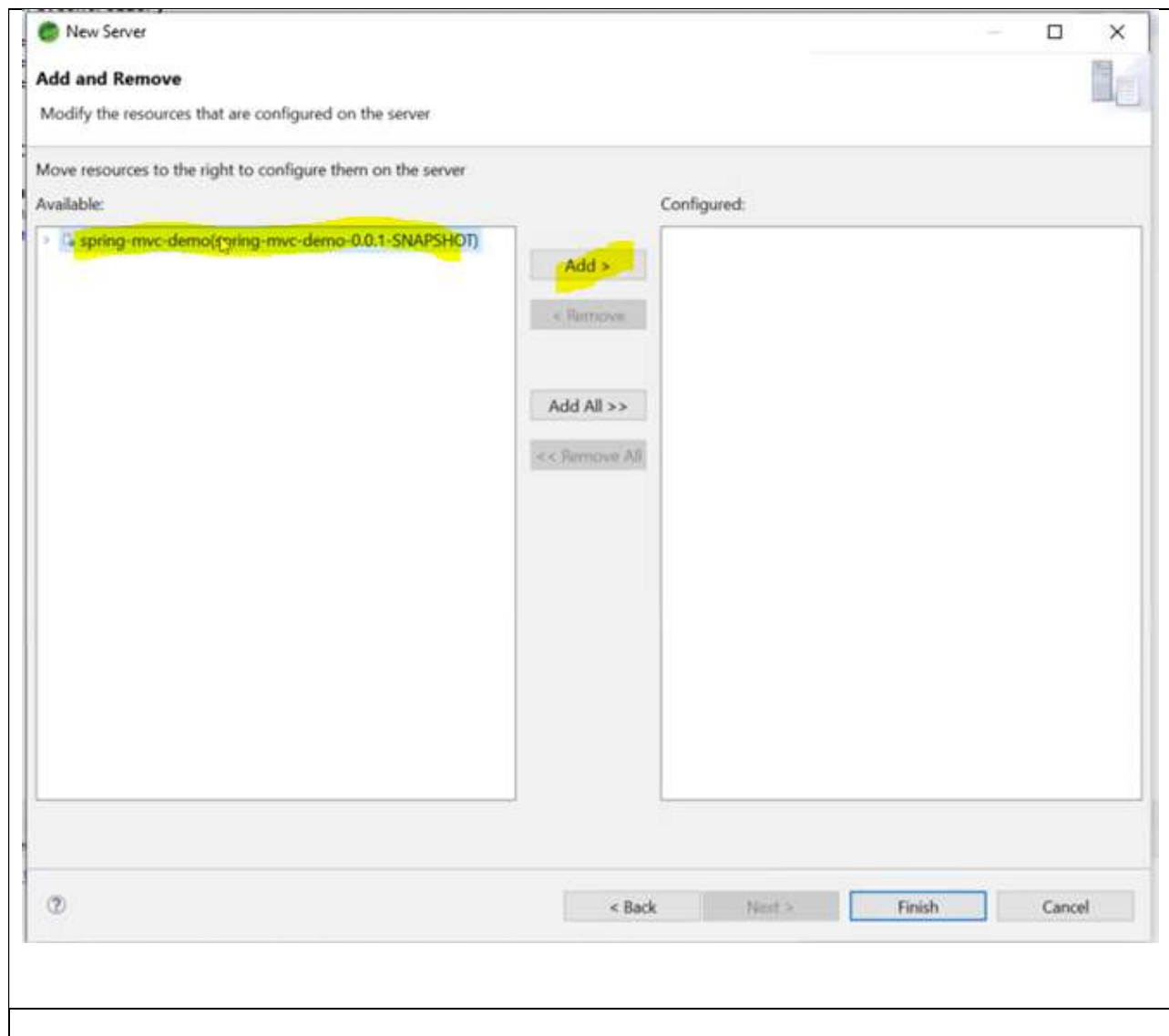


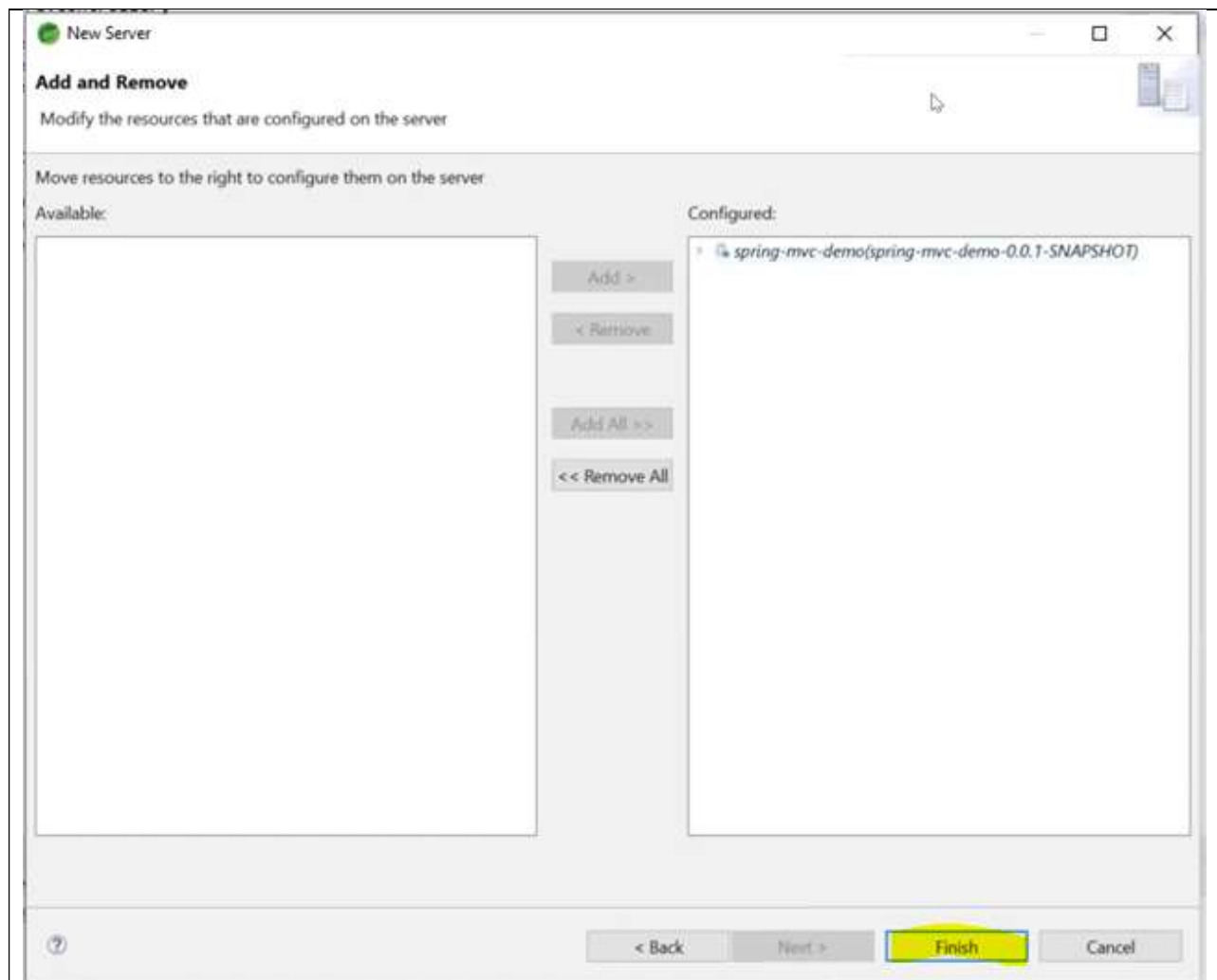




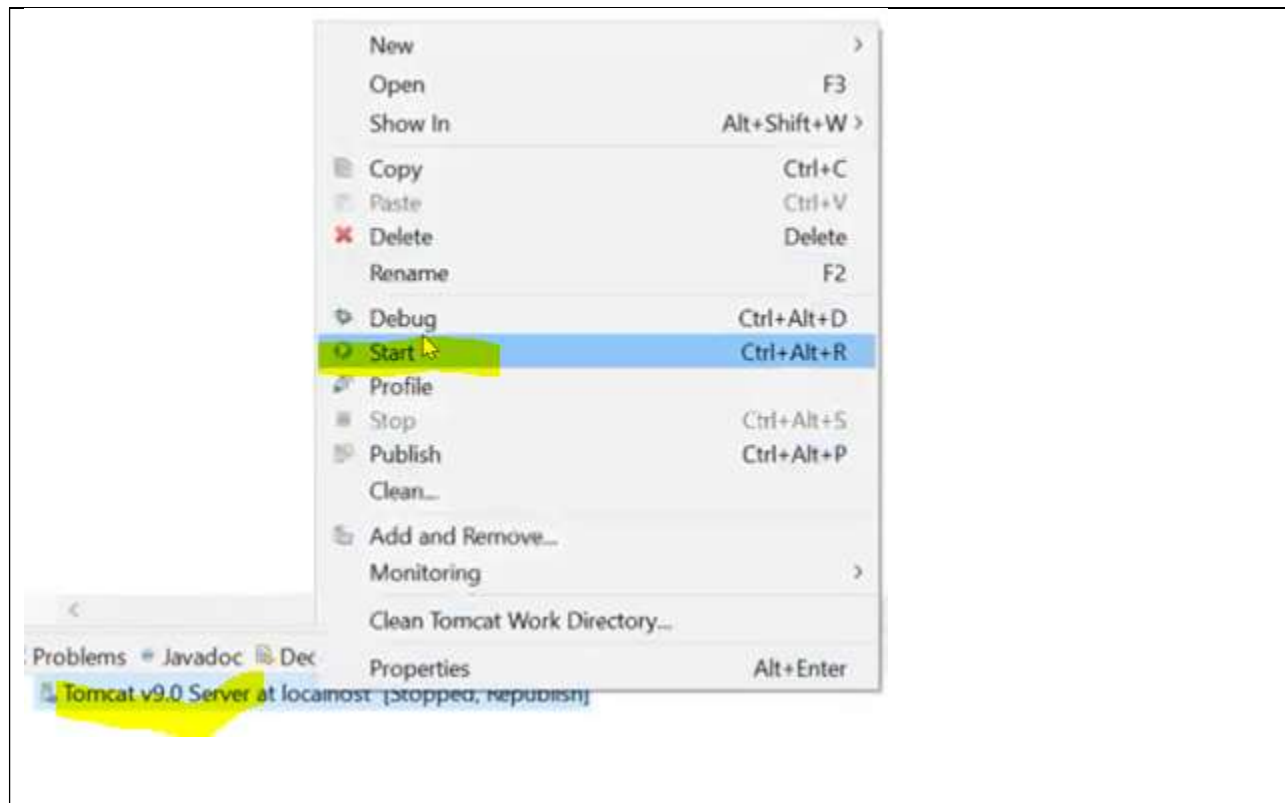




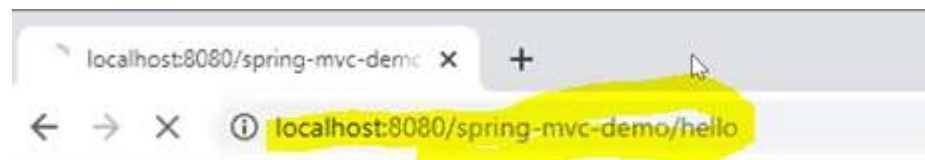




Right click on server then click start



Test server using the browser



localhost:8080/spring-mvc-with-orm-testing-demo/hello

Tomcat Server Setup Complete

<http://localhost:8080/spring-mvc-with-orm-testing-demo/hello>

<http://localhost:8080/spring-mvc-with-orm-testing-demo/hello>  
<artifactId>spring-mvc-with-orm-testing-demo</artifactId>  
<display-name>spring-mvc-with-orm-demo</display-name>

<http://localhost:8080/spring-mvc-with-orm-testing-demo/hello>

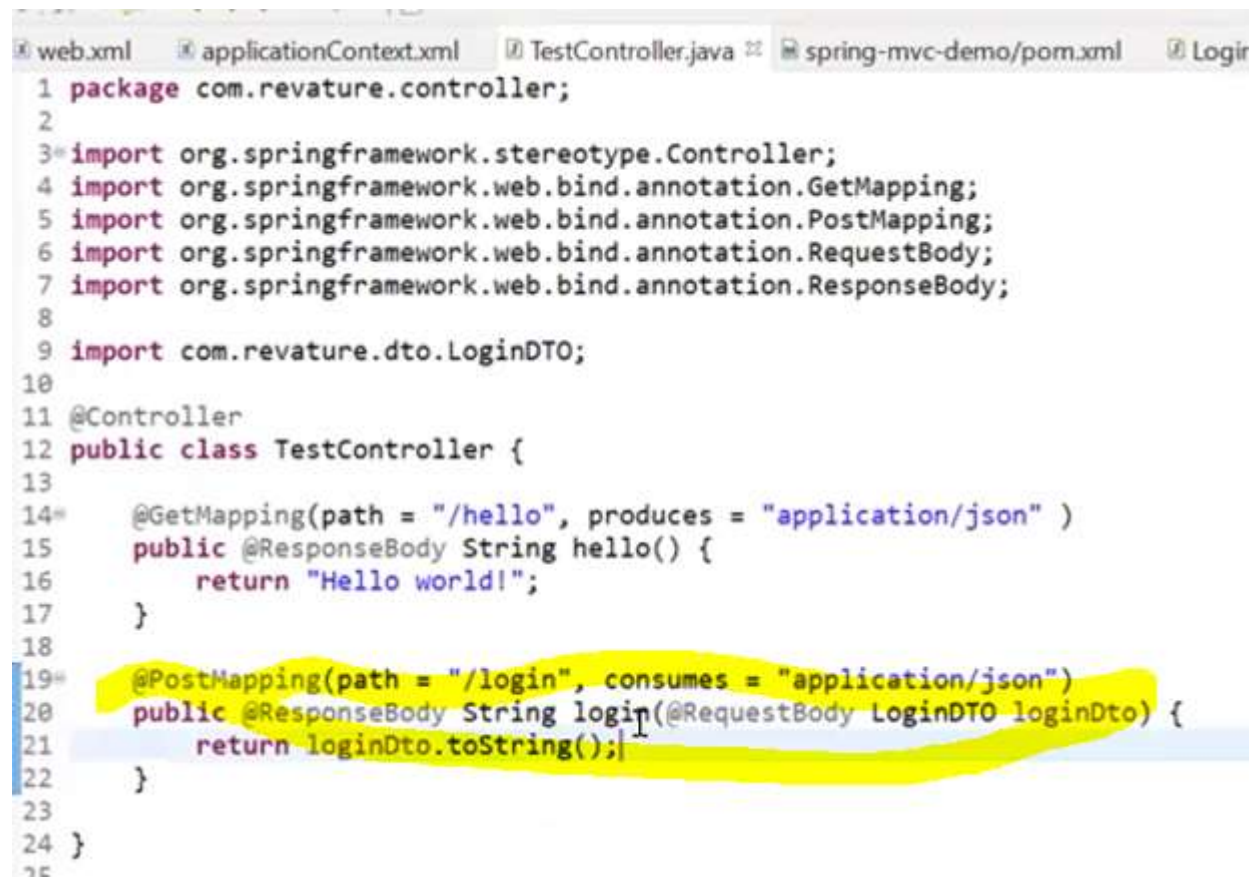
<http://localhost:8080/amt-mvc/hello>  
[http://localhost:8080/amt\\_user/tw8253](http://localhost:8080/amt_user/tw8253)  
[http://localhost:4201/amt\\_user/tw8253](http://localhost:4201/amt_user/tw8253)

<http://localhost:8080/spring-mvc/hello>

<http://localhost:8088/amt-mvc-with-orm-testing/hello>

amt-mvc-with-orm-testing

## Post Mapping



```
web.xml  applicationContext.xml  TestController.java  spring-mvc-demo/pom.xml  Login
1 package com.revature.controller;
2
3 import org.springframework.stereotype.Controller;
4 import org.springframework.web.bind.annotation.GetMapping;
5 import org.springframework.web.bind.annotation.PostMapping;
6 import org.springframework.web.bind.annotation.RequestBody;
7 import org.springframework.web.bind.annotation.ResponseBody;
8
9 import com.revature.dto.LoginDTO;
10
11 @Controller
12 public class TestController {
13
14     @GetMapping(path = "/hello", produces = "application/json" )
15     public @ResponseBody String hello() {
16         return "Hello world!";
17     }
18
19     @PostMapping(path = "/login", consumes = "application/json")
20     public @ResponseBody String login(@RequestBody LoginDTO loginDto) {
21         return loginDto.toString();
22     }
23
24 }
```

