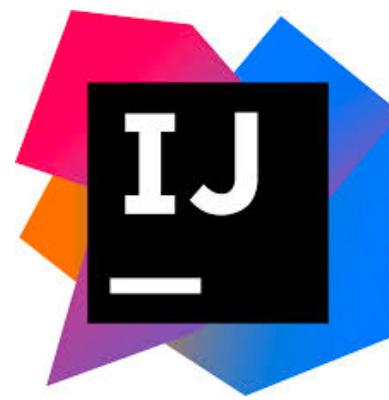


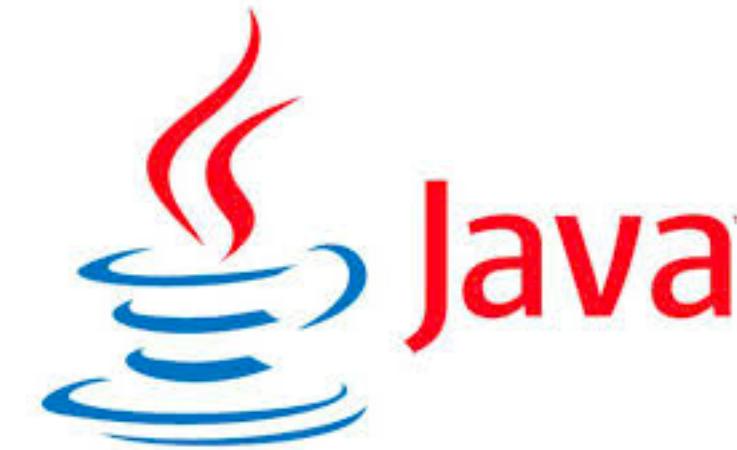
Client-Server App

(slides by Dr. K)



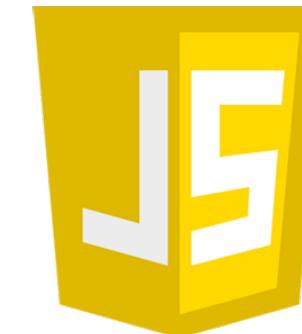
IntelliJ IDEA

server code



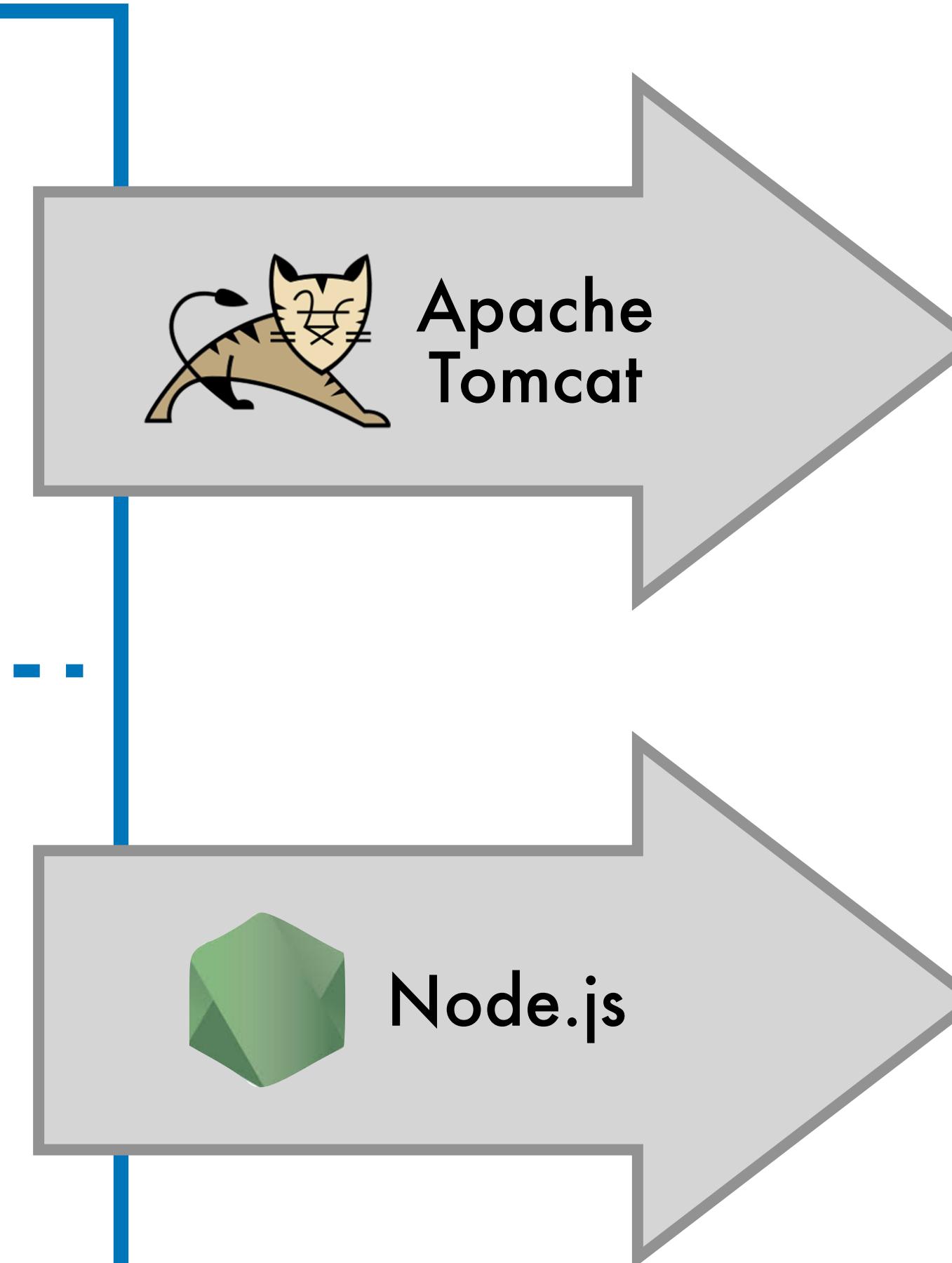
(code that runs on the server)

client code

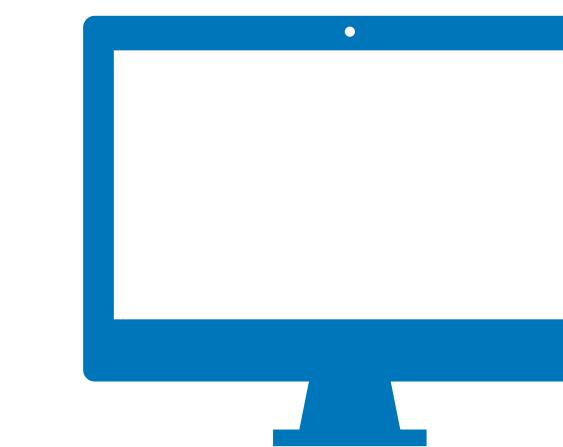


JavaScript

(code that runs on the browser)



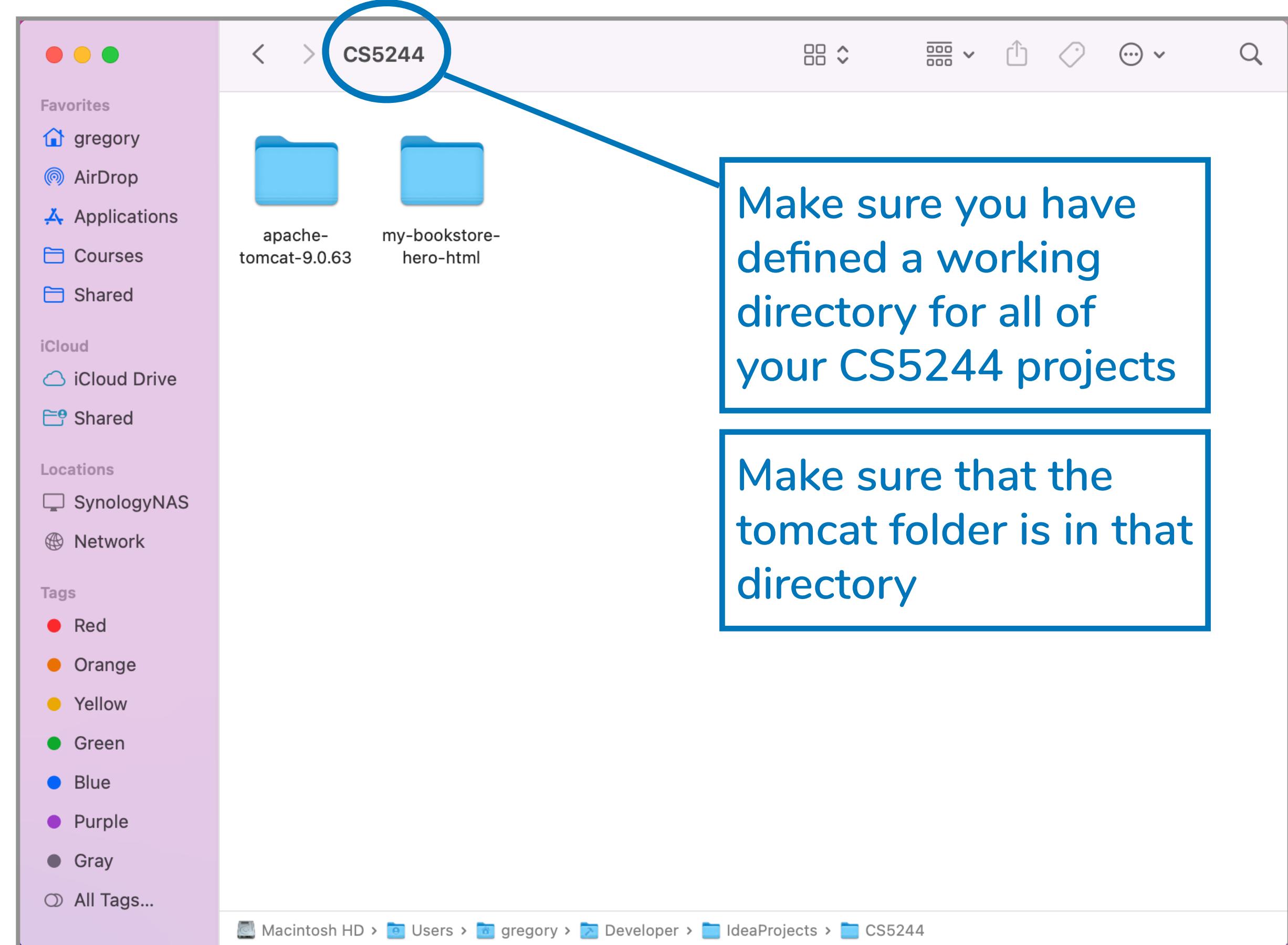
localhost:8080/ClientServer/



localhost:8081/

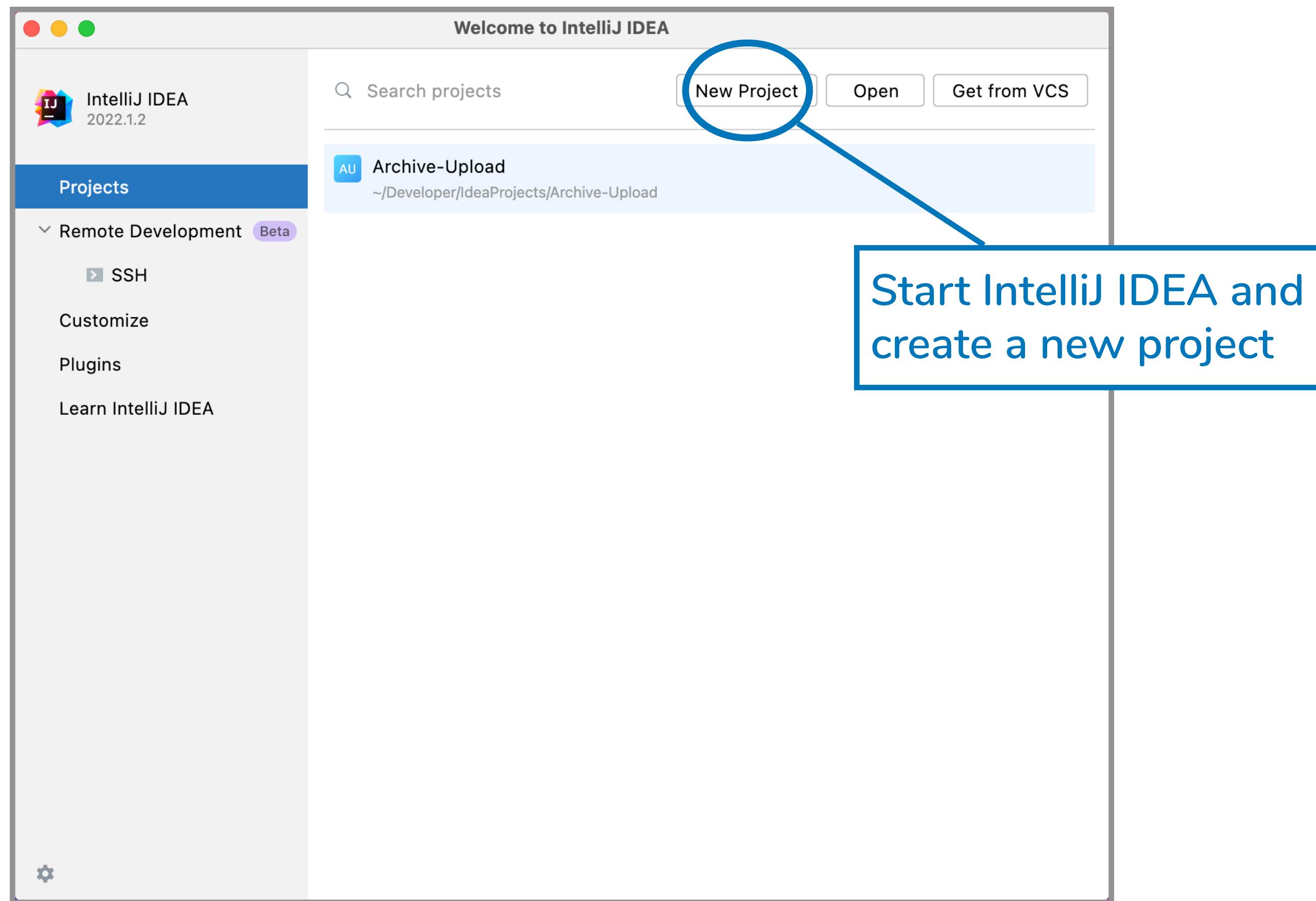
Before You Begin...

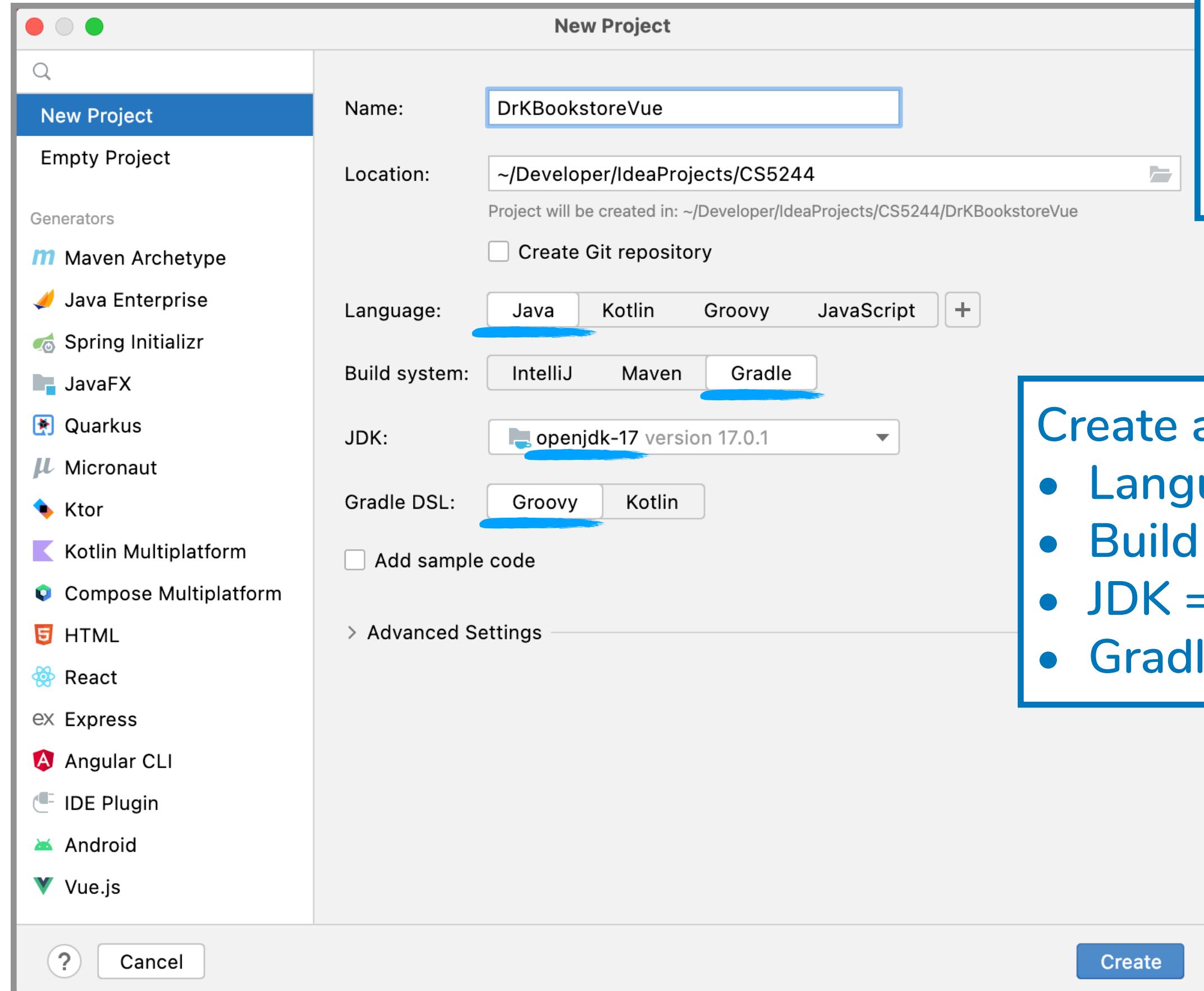
- Install [IntelliJ IDEA Ultimate](#) edition
- Create a **CS5244** working directory
- Install [Tomcat 9](#) in your working directory
- Install [Node.js](#) and [Vue-CLI](#)
- Install the Vue plugins for [Chrome](#) and [IntelliJ](#)



Make sure you have
defined a working
directory for all of
your CS5244 projects

Make sure that the
tomcat folder is in that
directory





Give the project a Name of [Name]BookstoreVue and ensure that it is in the CS5244 working directory

Create a Project with:

- Language = Java
- Build system = Gradle
- JDK = Java 17
- Gradle DSL = Groovy

```
DrKBookstoreVue – build.gradle (DrKBookstoreVue)
DrKBookstoreVue > build.gradle

Project Structure
DrKBookstoreVue ~/Developer/IdeaProjects/CS5244/
  .gradle
  .idea
  gradle
    src
      build.gradle
    gradlew
    gradlew.bat
    settings.gradle
External Libraries
Scratches and Consoles

build.gradle (DrKBookstoreVue) x
plugins {
    id 'java'
}

group 'org.example'
version '1.0-SNAPSHOT'

repositories {
    mavenCentral()
}

dependencies {
    testImplementation 'org.junit.jupiter:junit-jupiter-api:5.8.1'
    testRuntimeOnly 'org.junit.jupiter:junit-jupiter-engine:5.8.1'
}

test {
    useJUnitPlatform()
}
```

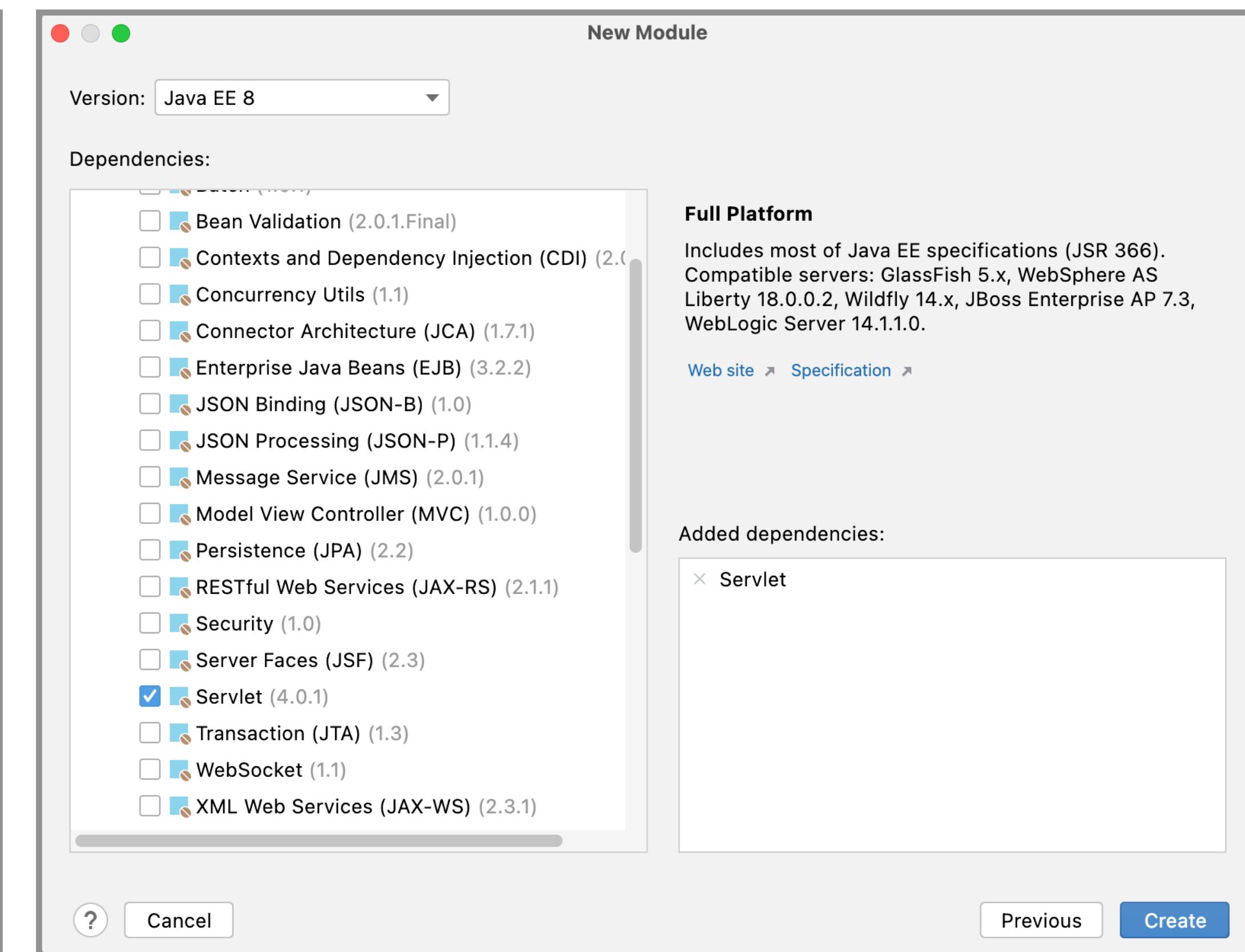
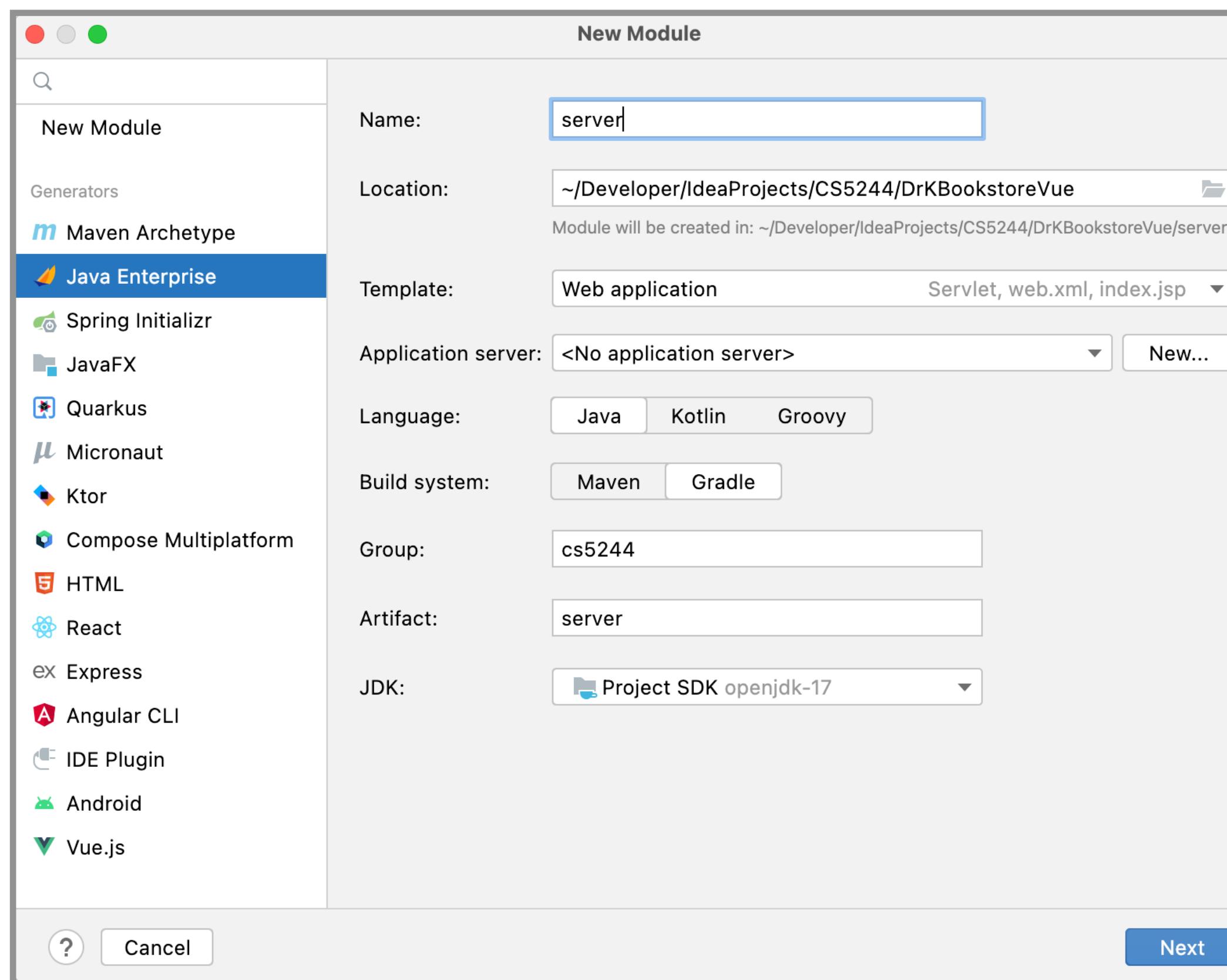
Once the project is created, select ***all*** the text in the build.gradle file and ***delete*** it. (But keep the empty file.)

Also, delete the entire "src" directory. We do not need it. IntelliJ will warn you that you are deleting all the files inside. This is fine.

Right-click on the project directory ([Name]BookstoreVue) to create a new module.

Name: server
Location: (inside project folder)
Template: Web application
Language: Java
Build system: Gradle
JDK: Java 17

Select "Servlet" for Dependencies

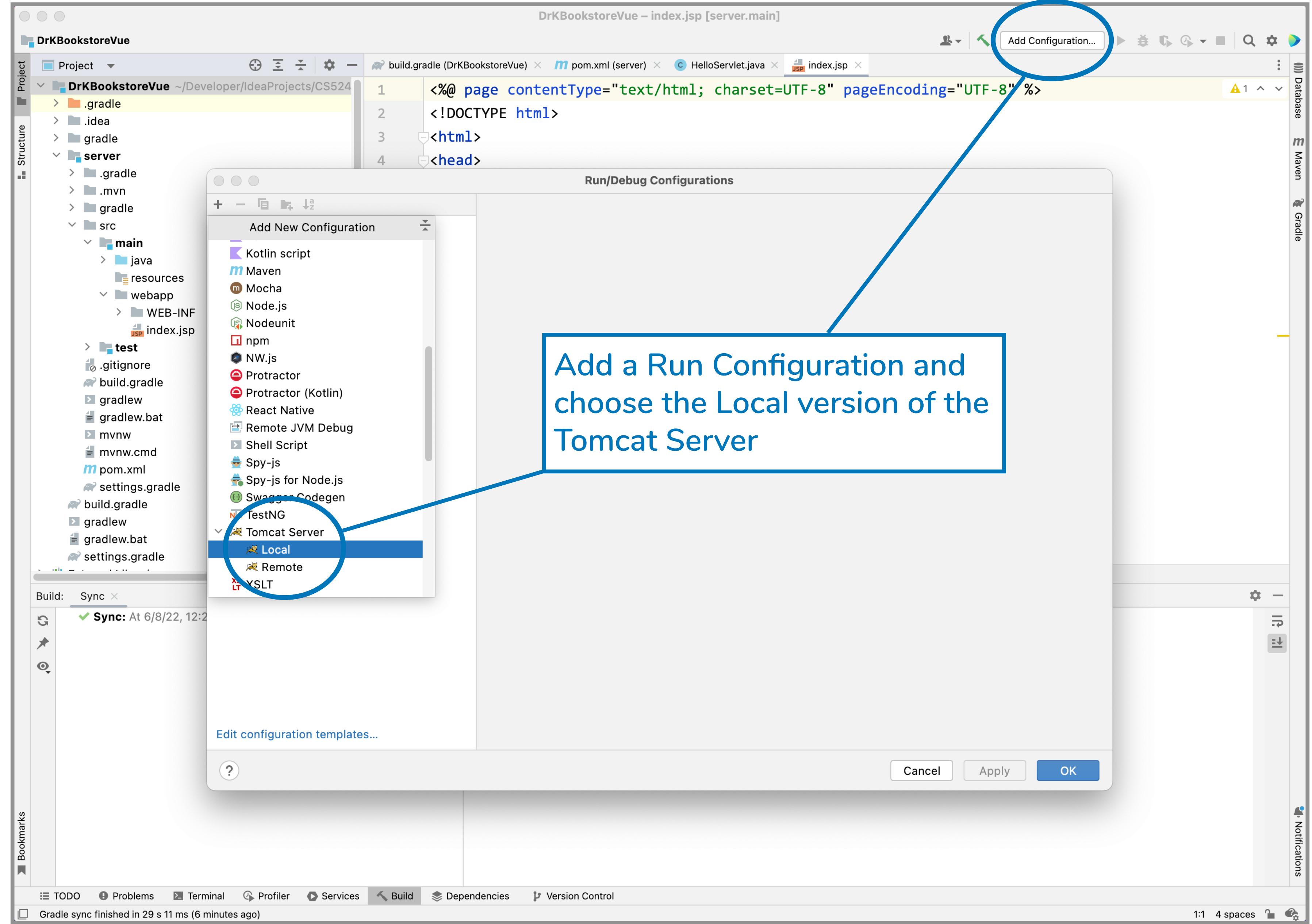


The screenshot shows the IntelliJ IDEA interface with the following details:

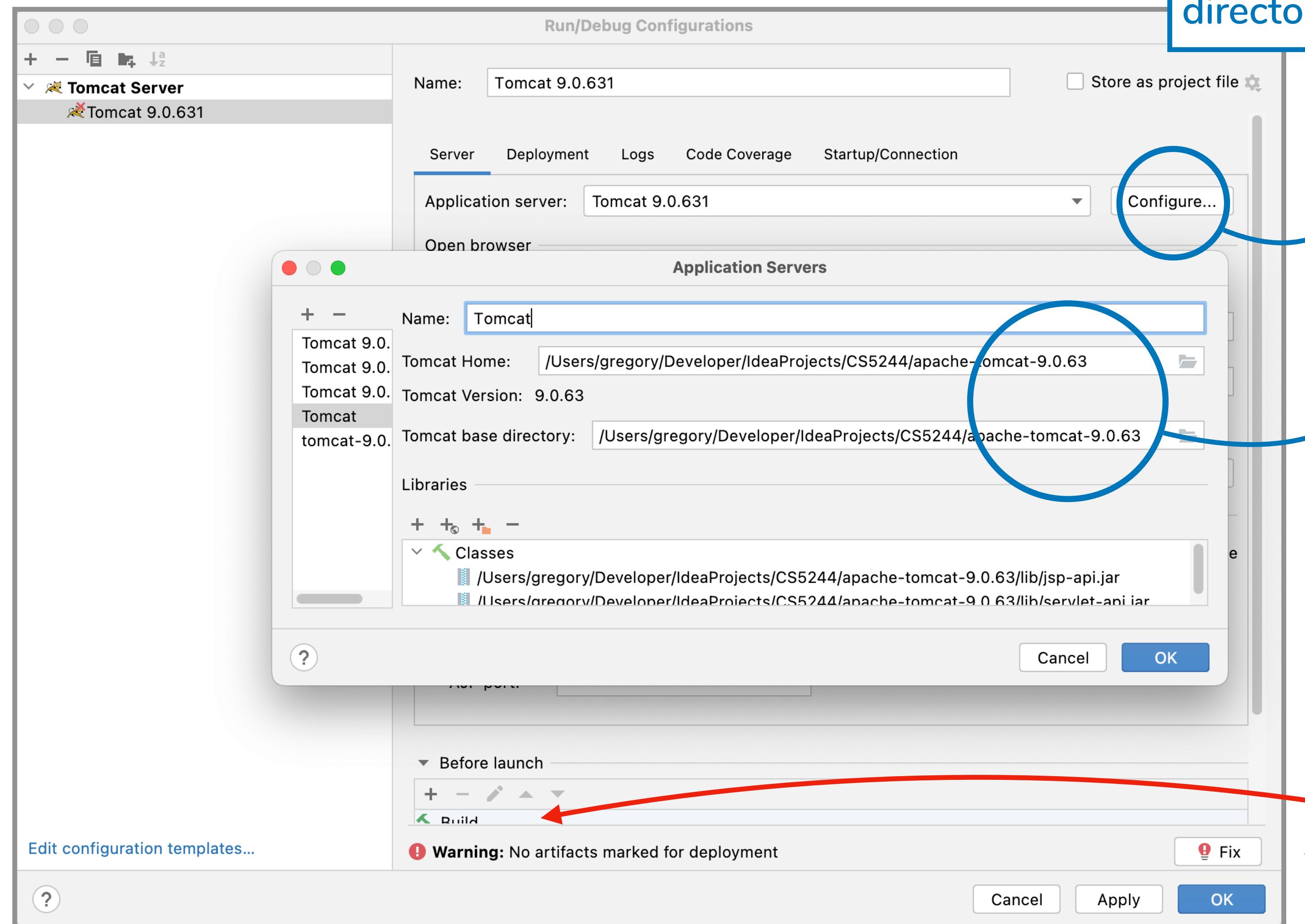
- Project Structure:** The left sidebar displays the project structure for "DrKBookstoreVue". A blue circle highlights the "src/main/webapp" directory, which contains "index.jsp".
- Code Editor:** The main window shows the content of "index.jsp". The code is as follows:

```
<%@ page contentType="text/html; charset=UTF-8" pageEncoding="UTF-8" %>
<!DOCTYPE html>
<html>
<head>
    <title>DrKBookstoreVue</title>
</head>
<body>
    <h1><%= "Welcome!" %>
    <br/>
    <a href="hello-servlet">Hello Servlet</a>
</body>
</html>
```

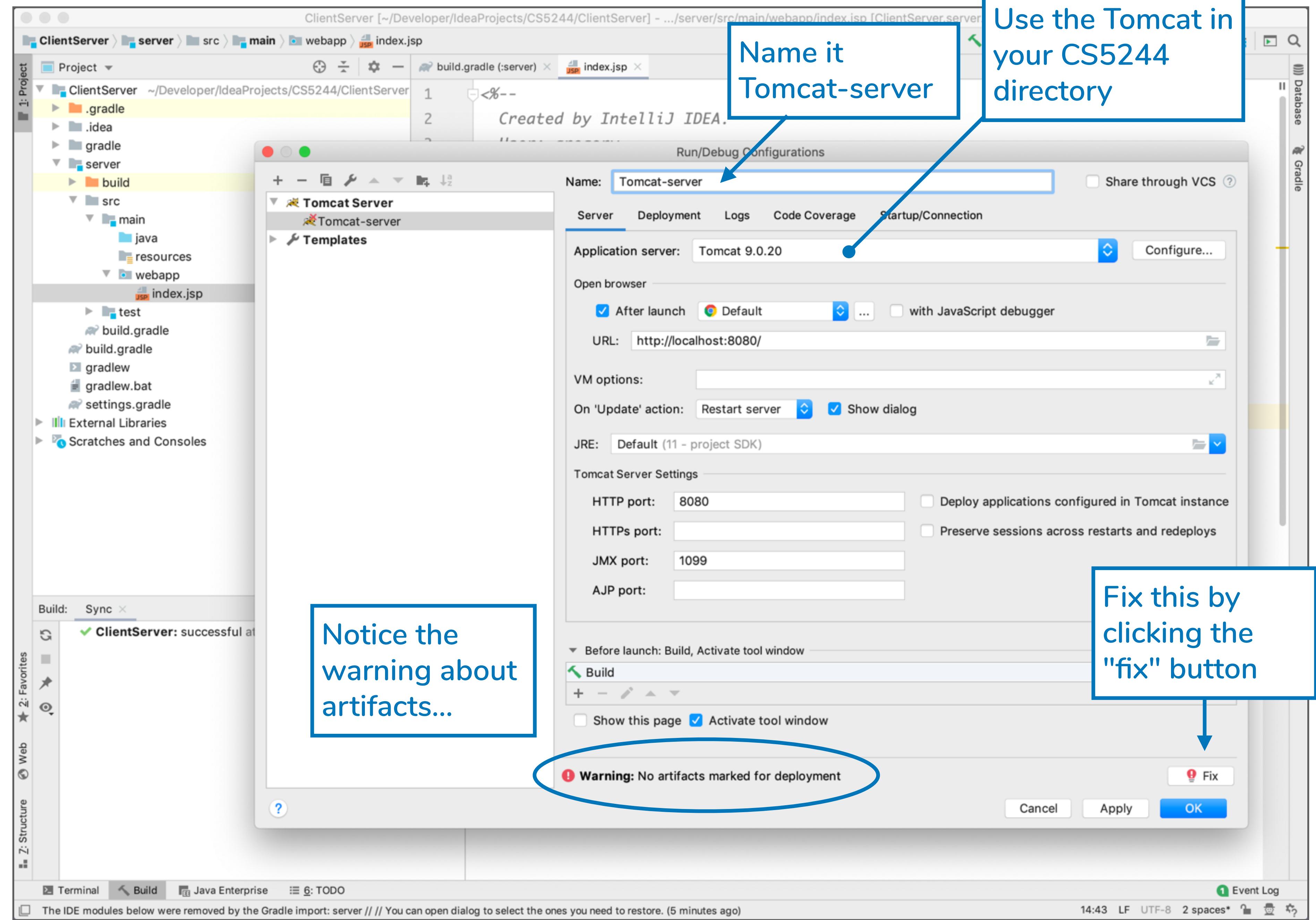
The server should have a directory called webapp with an index.jsp file. There should also be a WEB-INF directory with a web.xml file.

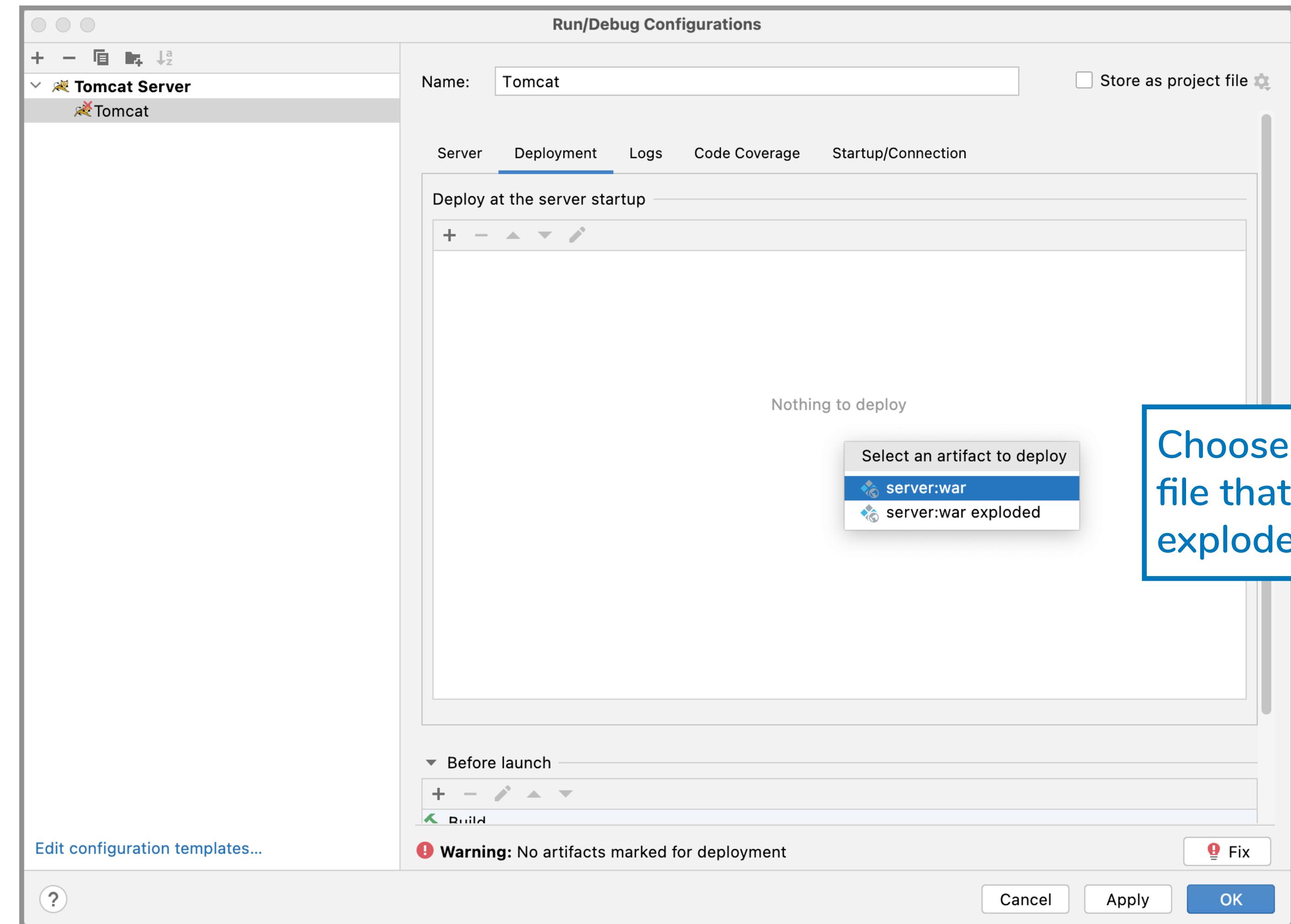


Configure the server by choosing the Tomcat version in your CS5244 directory

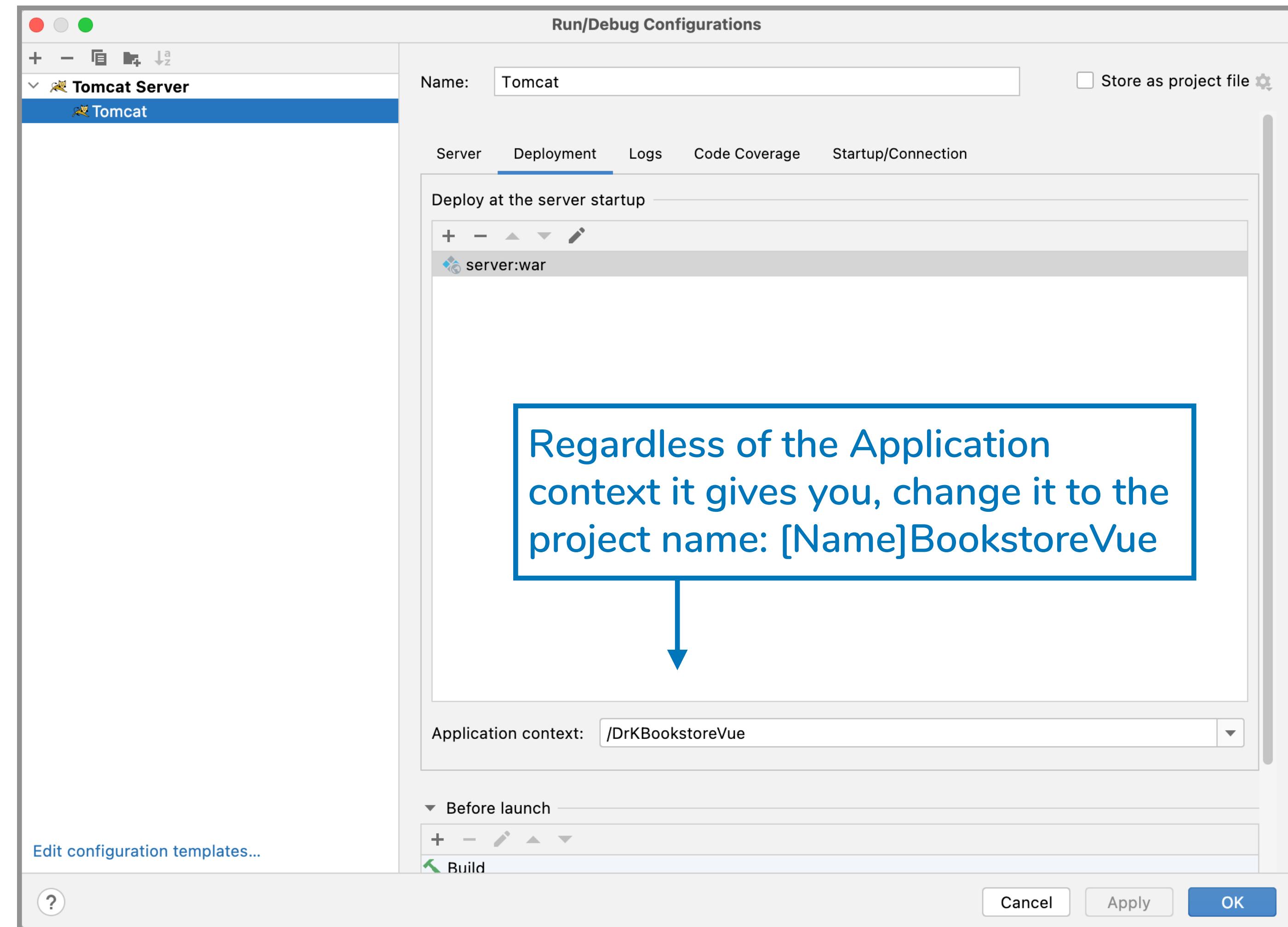


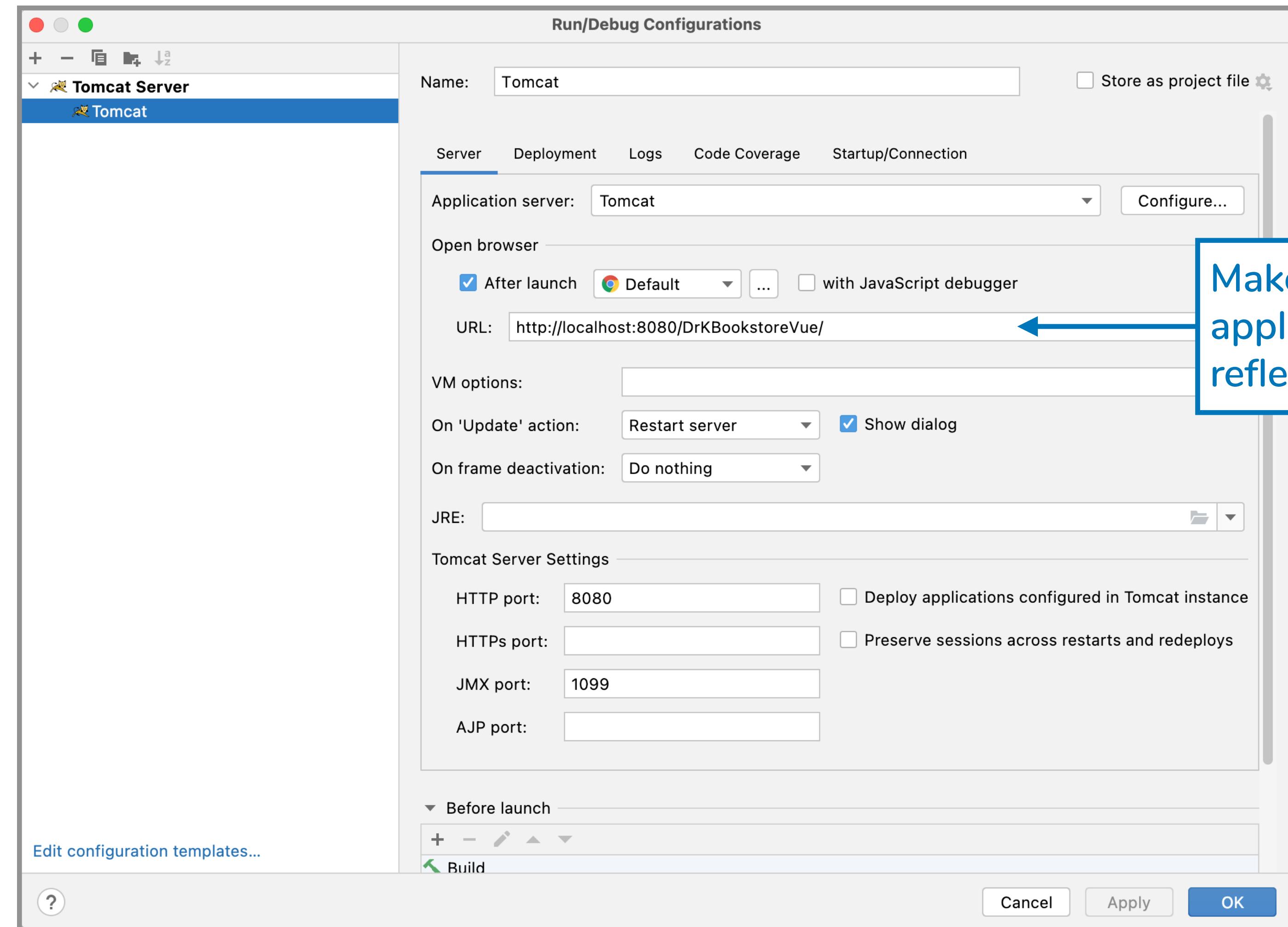
Click on the button to fix the warning





Choose the WAR
file that is *not*
exploded



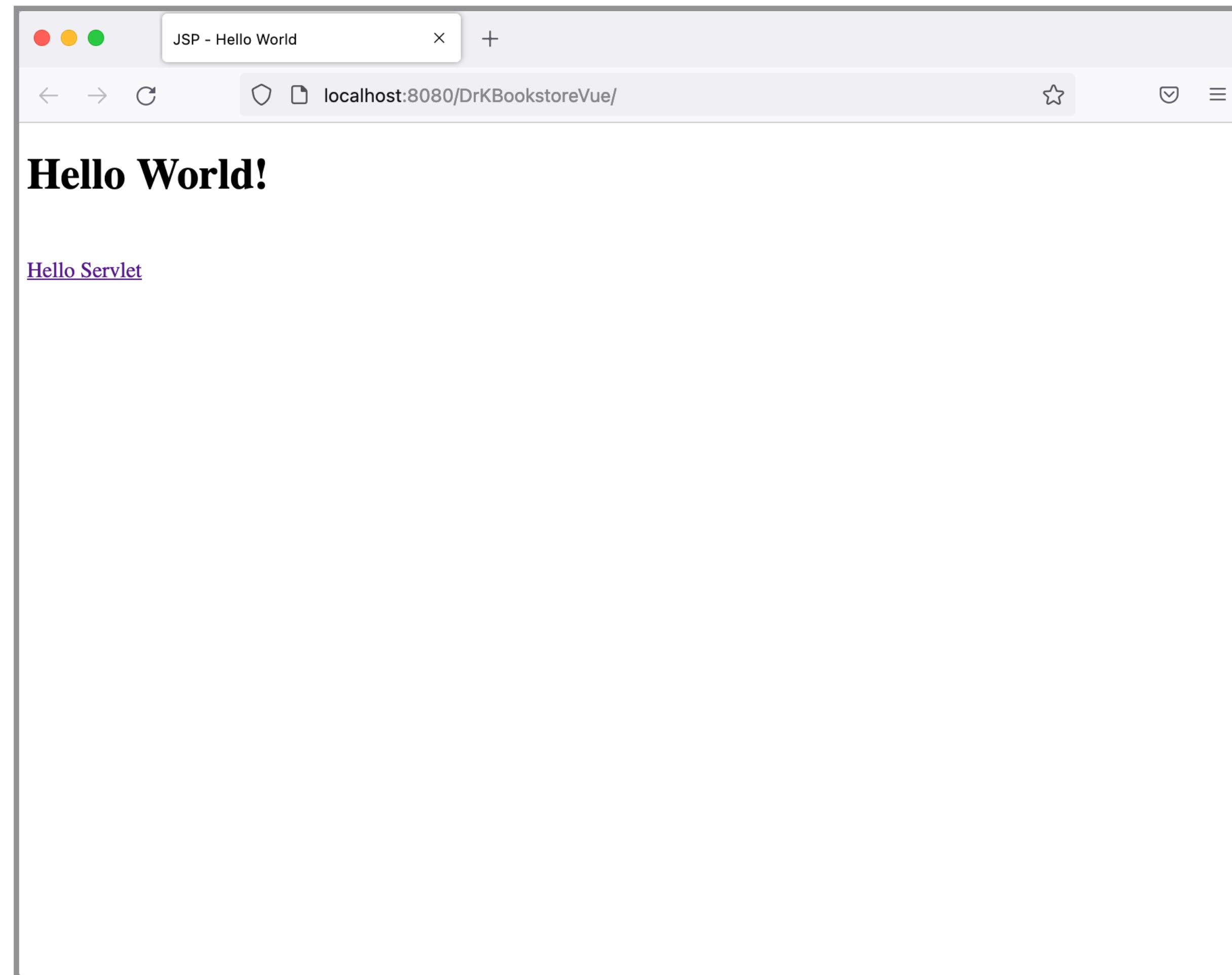


Make sure the
application context is
reflected in the URL

Run the application by selecting the Tomcat-server run configuration and clicking the green arrow (➤) next to it

This is what you should see in your browser

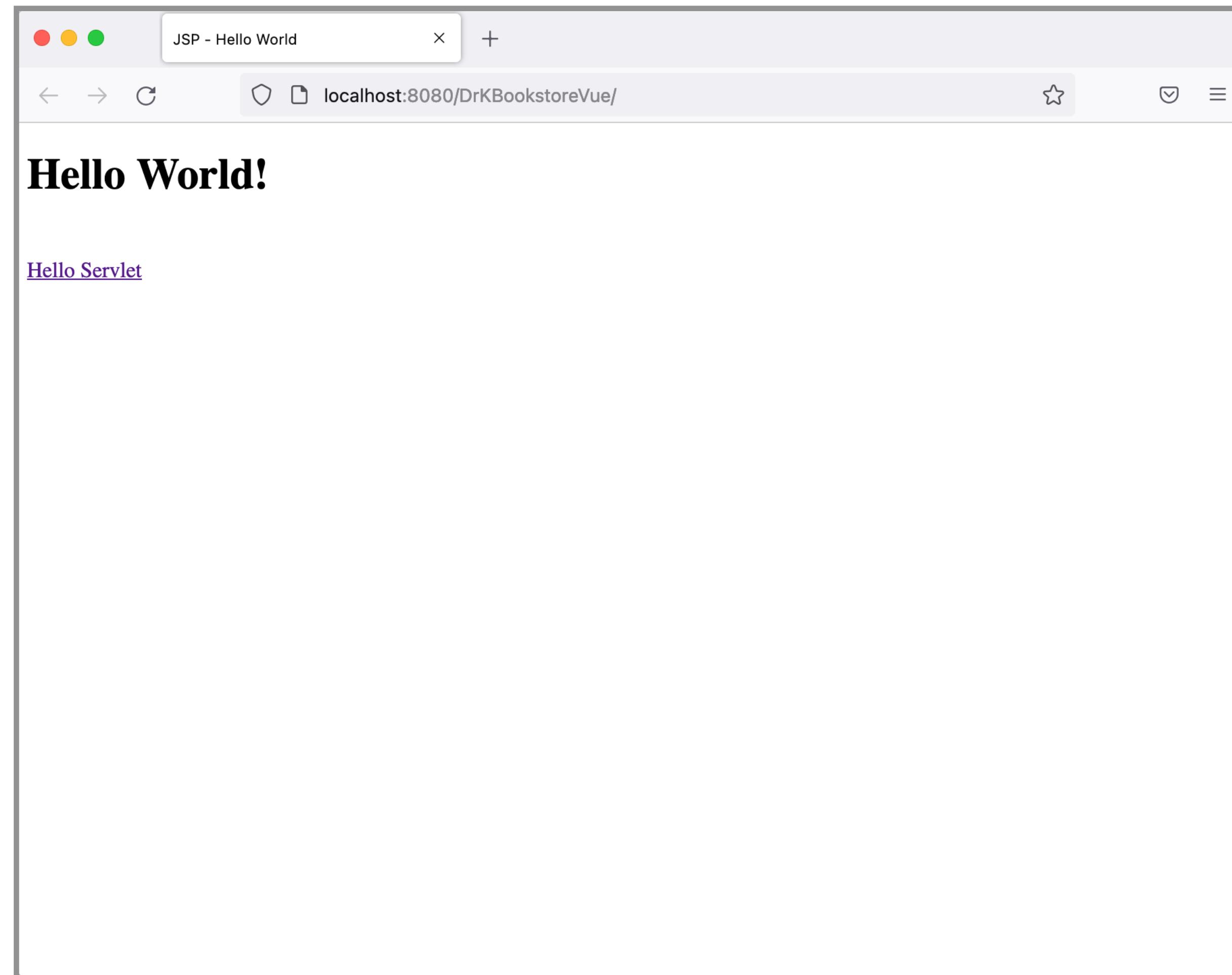
Tomcat is serving your Java-based application



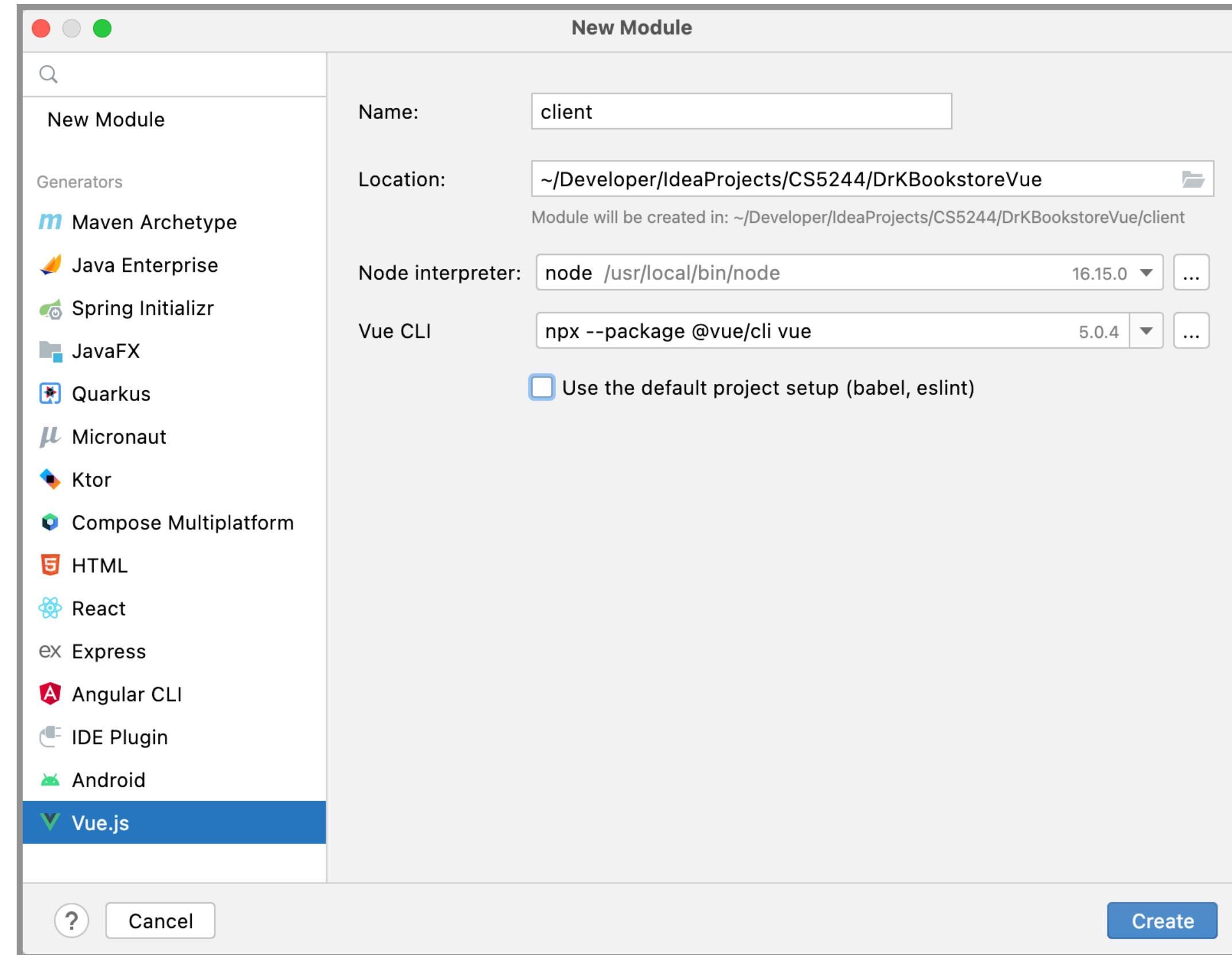
Run the application by selecting the Tomcat-server run configuration and clicking the green arrow (➤) next to it

This is what you should see in your browser

Tomcat is serving your Java-based application



Create a new module again in your project. This time we will create the "client" module using Vue.js



Make sure "Use default project setup" is *not* checked.

The screenshot shows an IDE interface with the following details:

- Project Structure:** The project is named "DrKBookstoreVue". It contains a "server" directory which includes "build.gradle", "client.iml", "gradlew", "gradlew.bat", and "settings.gradle".
- Code Editor:** The "index.jsp" file is open, displaying JSP code. The code includes a welcome message and a link to a servlet.
- Run Dialog:** A "Generating client" task is running, and the "Vue CLI v5.0.4" configuration is shown. It asks the user to pick a preset or manually select features. The "Manually select features" option is highlighted with a blue arrow pointing to a callout box.
- Callout Box:** A blue box with a white background and a thin blue border contains the text: "Choose to manually select the features."

```
<%@ page contentType="text/html; charset=UTF-8" pageEncoding="UTF-8" %>
<!DOCTYPE html>
<html>
<head>
    <title>DrKBookstoreVue</title>
</head>
<body>
    <h1><%= "Welcome!" %></h1>
    <br/>
    <a href="hello-servlet">Hello Servlet</a>
</body>
</html>
```

Run: Generating client

Vue CLI v5.0.4

? Please pick a preset:

- prettier-router-vuex ([Vue 2] router, vuex, eslint)
- Default ([Vue 3] babel, eslint)
- Default ([Vue 2] babel, eslint)

› Manually select features

All files are up-to-date (2 minutes ago)

Run: Generating client

```
? Please pick a preset: Manually select features
? Check the features needed for your project: (Press <space> to select, <a> to toggle all, <i> to invert selection, and <enter> to proceed)
> Babel
  ○ TypeScript
  ○ Progressive Web App (PWA) Support
  ● Router
  ○ Vuex
  ○ CSS Pre-processors
  ● Linter / Formatter
  ○ Unit Testing
  ○ E2E Testing
```

Bookmarks

Run TODO Problems Terminal Profiler Services Build Dependencies Version Control

All files are up-to-date (3 minutes ago) 13:8 LF UTF-8 4 spaces 🔒 ⚙️

For this project, do *not* use Babel. Select the Router and select the Linter

Run: Generating client

```
? Please pick a preset: Manually select features
? Check the features needed for your project: Router, Linter
? Choose a version of Vue.js that you want to start the project with
  3.x
> 2.x
```

Choose Vue 2.x for this project

Run: Generating client

```
? Please pick a preset: Manually select features
? Check the features needed for your project: Router, Linter
? Choose a version of Vue.js that you want to start the project with 2.x
? Use history mode for router? (Requires proper server setup for index fallback in production) Yes
? Pick a linter / formatter config: Prettier
? Pick additional lint features: Lint on save
? Where do you prefer placing config for Babel, ESLint, etc.? In dedicated config files
? Save this as a preset for future projects? (y/N)
```

Bookmarks

Run TODO Problems Terminal Profiler Services Build Dependencies Version Control

All files are up-to-date (7 minutes ago) 13:8 LF UTF-8 4 spaces 🔒 ⚙️

Choose Prettier for the Linter, Lint on save, dedicated config files, and don't save the preset

DrKBookstoreVue – index.jsp [server.main]

Project DrKBookstoreVue ~/Developer/IdeaProjects/CS5244/

Structure

build.gradle (DrKBookstoreVue) pom.xml (server) HelloServlet.java index.jsp

1 <%@ page contentType="text/html; charset=UTF-8" pageEncoding="UTF-8" %>
2 <!DOCTYPE html>
3 <html>
4 <head>
5 <title>DrKBookstoreVue</title>
6 </head>
7 <body>
8 <h1><%= "Welcome!" %>
9 </h1>
10

11 Hello Servlet
12 </body>
13 </html>

Run: Generating client

Running completion hooks...

Generating README.md...

Successfully created project client.

Get started with the following commands:

\$ npm run serve

Done

Once the client is created, you can run it using: npm run serve

Database Maven Gradle

Notifications

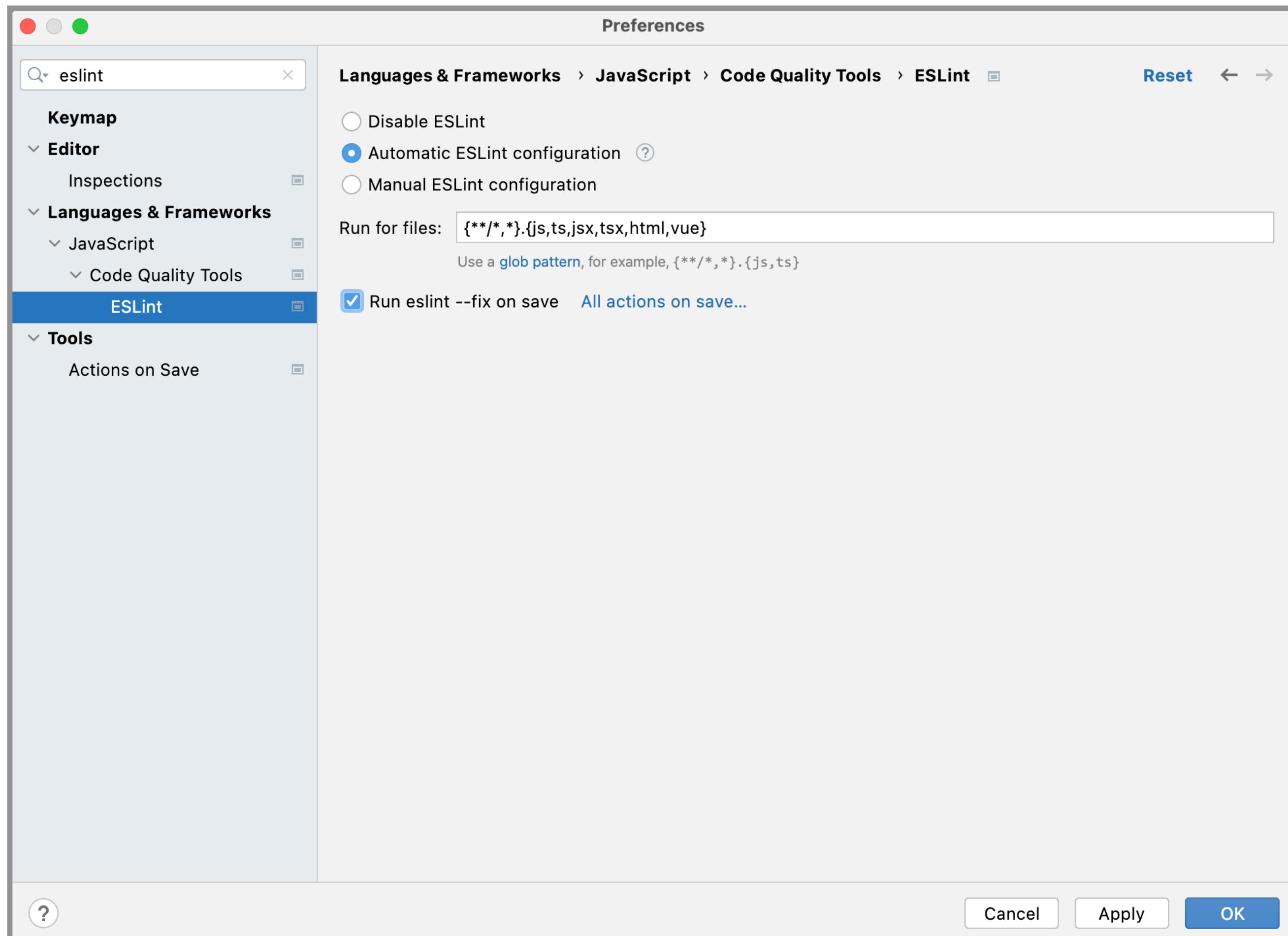
Run TODO Problems Terminal Profiler Services Build Dependencies Git

ESLint is enabled: 'eslint' is listed in package.json. // Disable ESLint (moments ago)

13:8 LF UTF-8 4 spaces master

```
1 module.exports = {
2   publicPath: "/DrKBookstoreVue/",
3 };
4 |
```

Create a JavaScript file in your client called vue.config.js. Set the publicPath to your project name. This is similar to setting the application context on the server side.



Go into your preferences, search for eslint, and click the checkbox next to fix-on-save. This will automatically clean up your code every time you save a file.

The screenshot shows an IDE interface with the following details:

- Project Structure:** The project is named "DrKBookstoreVue". It contains a "server" directory which is expanded, showing files like ".gradle", ".idea", "client", "gradle", and "server".
- Code Editor:** The file "index.jsp" is open. The code is as follows:

```
<%@ page contentType="text/html; charset=UTF-8" pageEncoding="UTF-8" %>
<!DOCTYPE html>
<html>
<head>
<title>DrKBookstoreVue</title>
</head>
<body>
<h1><%= "Welcome!" %></h1>
<br/>
<a href="hello-servlet">Hello Servlet</a>
</body>
</html>
```

- Terminal:** A "Local" terminal window is open, displaying the following error message from oh-my-zsh:

```
[oh-my-zsh] the write permission of "group" and "others" and making sure that the
[oh-my-zsh] owner of these directories is either root or your current user.
[oh-my-zsh] The following command may help:
[oh-my-zsh]     compaudit | xargs chmod g-w,o-w

[oh-my-zsh] If the above didn't help or you want to skip the verification of
[oh-my-zsh] insecure directories you can set the variable ZSH_DISABLE_COMPFIX to
[oh-my-zsh] "true" before oh-my-zsh is sourced in your zshrc file.

→ DrKBookstoreVue cd client
→ client git:(master) npm run serve
```

To run the client, open a terminal,
"cd" into the client directory
(that's important!), and then type
npm run serve

DrKBookstoreVue – index.jsp [server.main]

DrKBookstoreVue

Project Structure

DrKBookstoreVue ~/Developer/IdeaProjects/CS5244/

- > .gradle
- > .idea
- > client
- > gradle
- > server
 - build.gradle
 - client.iml
 - gradlew
 - gradlew.bat
 - settings.gradle
- > External Libraries
- > Scratches and Consoles

index.jsp

```
<%@ page contentType="text/html; charset=UTF-8" pageEncoding="UTF-8" %>
<!DOCTYPE html>
<html>
<head>
    <title>DrKBookstoreVue</title>
```

When you run the command, Node.js serves your client-side code using port 8081

```
13</body>
</html>
```

The default port for Node is 8080. However, the Tomcat server is already running the server-side code on 8080

So Node adjusts by incrementing the port number

Terminal: Local +

```
App running at:
- Local: http://localhost:8081/
- Network: http://192.168.1.5:8081/
```

Note that the development build is not optimal. To create a production build, run `npm run build`.

Click on the link to see the app in your browser

Debug Application

Database

Maven

Gradle

Notifications

Run TODO Problems Terminal Profiler Services Build Dependencies Git

ESLint is enabled: 'eslint' is listed in package.json. // Disable ESLint (2 minutes ago)

13:8 LF UTF-8 4 spaces master

DrKBookstoreVue

localhost:8080/DrKBookstoreVue/

Welcome!

[Hello Servlet](#)

The default text said "Hello, World!". Can you figure out how to change it?

The server-side code is being served by Tomcat on URL localhost:8080/ClientServer

client

localhost:8081/DrKBookstoreVue/

[Home](#) | [About](#)



Welcome to Your Vue.js App

For a guide and recipes on how to configure / customize this project, check out the [vue-cli documentation](#).

Installed CLI Plugins

[router](#) [eslint](#)

Essential Links

[Core Docs](#) [Forum](#) [Community Chat](#) [Twitter](#) [News](#)

Ecosystem

[vue-router](#) [vuex](#) [vue-devtools](#) [vue-loader](#) [awesome-vue](#)

The client-side code is being served by Node.js on URL localhost:8081