

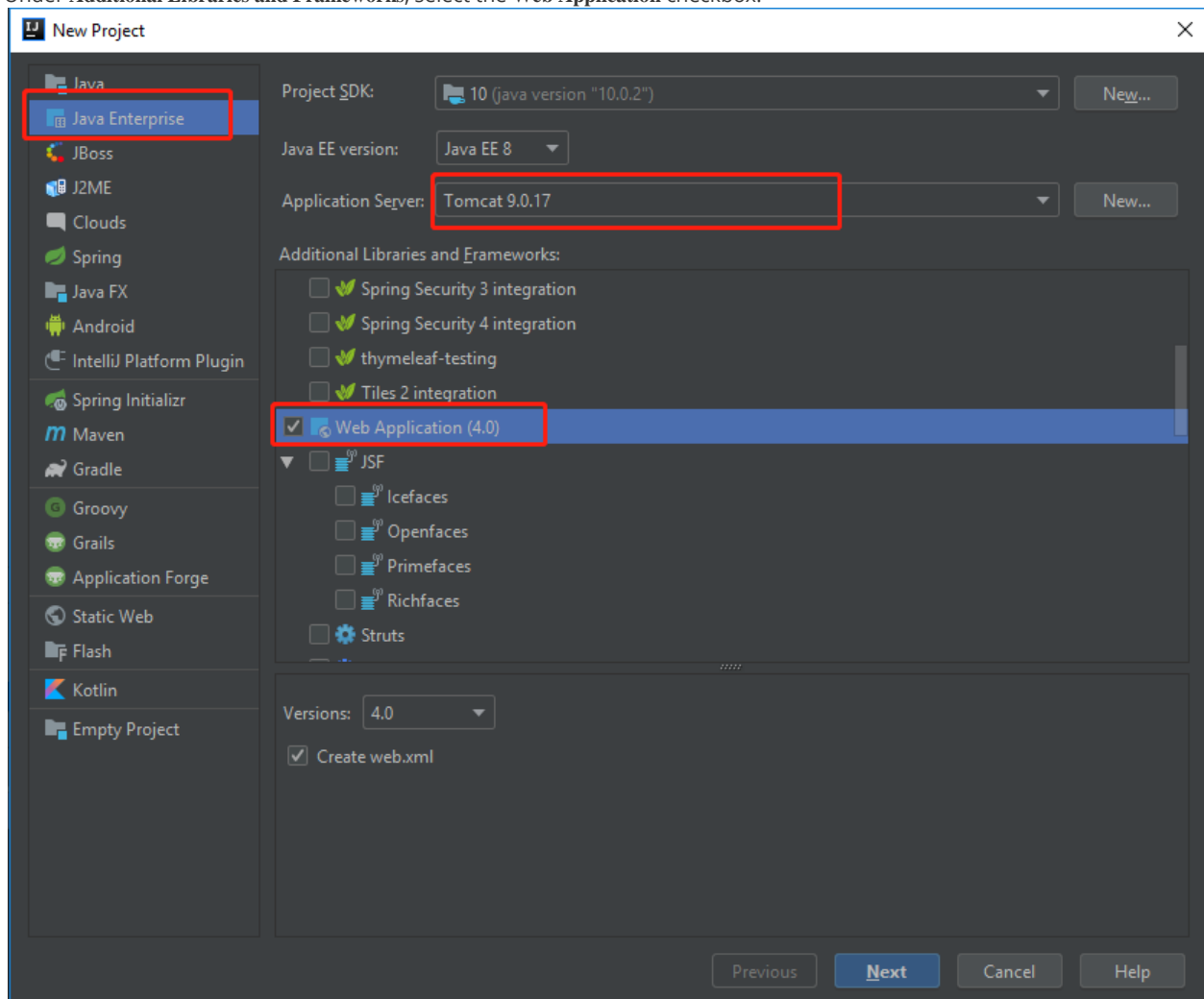
Set up IntelliJ With Tomcat

A.Download Tomcat and install. I use Tomcat 9

B.Create Dynamic Web Content Project(Plain Servlet & JSP)

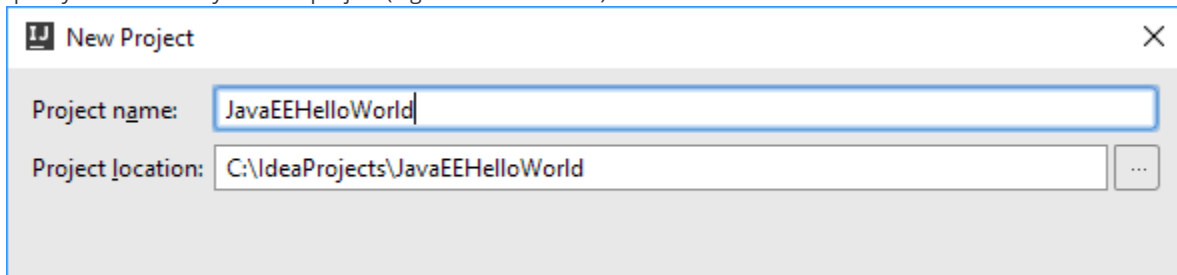
Creating a project

1. Click **Create New Project** on the Welcome screen, or select **File | New | Project**.
The **New Project** wizard opens.
2. In the left-hand pane, select **Java Enterprise**.
3. Specify the **JDK** that you are going to use (the **Project SDK** field): select one from the list, click **New** and select the JDK installation folder, or click **Download JDK**.
4. Specify your application server. (We'll use Tomcat Server.)
5. Under **Additional Libraries and Frameworks**, select the **Web Application** checkbox.



Click Next.

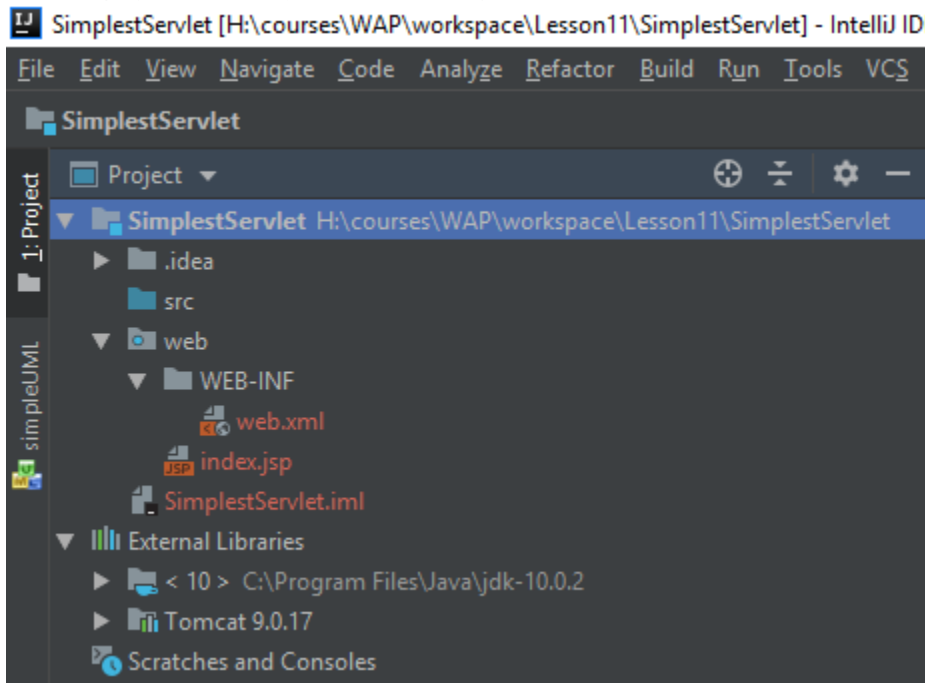
- Specify the name for your new project (e.g. *JavaEEHelloWorld*).



Click **Finish** and wait while IntelliJ IDEA is creating the project.

Exploring the project structure

When the project is created, you'll see something similar to this in the **Project** tool window.



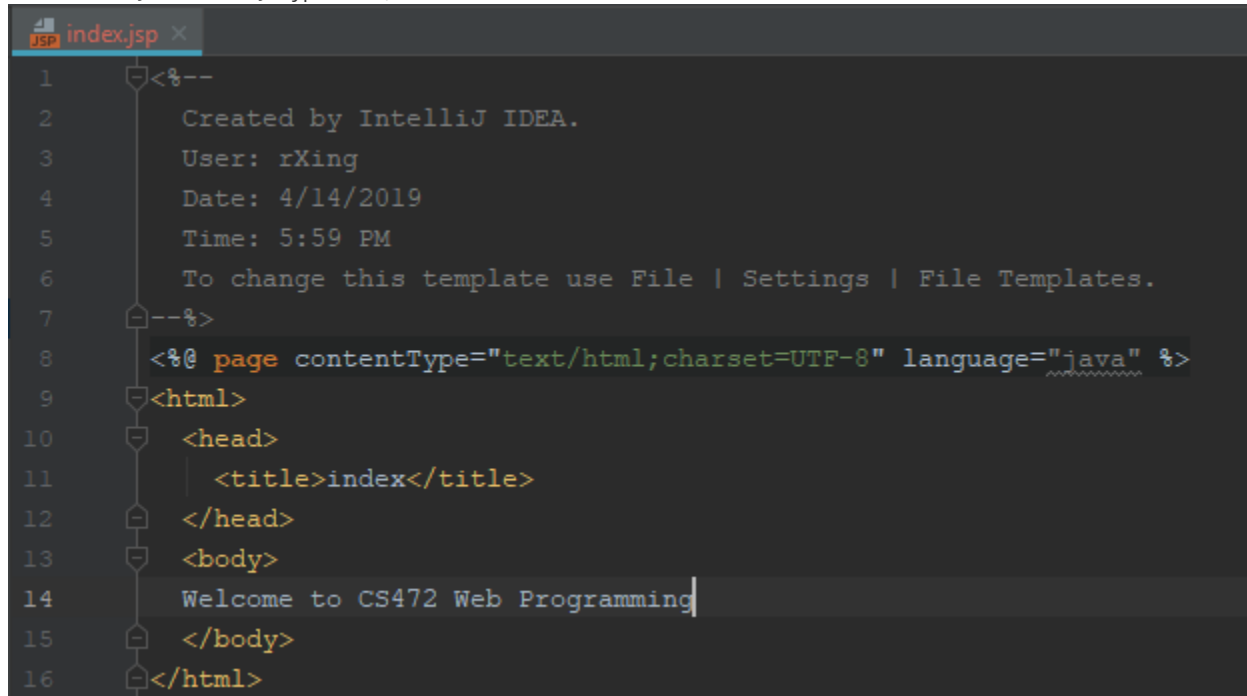
- SimplestServlet** is a module folder (which in this case coincides with the project folder). The **.idea** folder and the file **SimplestServlet.iml** contain configuration data for your project and module respectively. The folder **src** is for your Java source code. The folder **web** is for the web part of your application. At the moment this folder contains the deployment descriptor **WEB-INF/web.xml** and the file **index.jsp** intended as a starting page of your application.
- External Libraries** include your JDK and the JAR files for working with Tomcat.

Developing source code

Our application will be a single JSP page application. Its only function will be to output the text *Hello, World!*

- Open **index.jsp** for editing: select the file in the **Project** tool window and press F4.


- Between `<body>` and `</body>` type Hello, World!

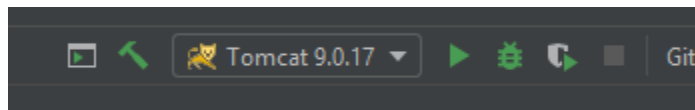


```
1  <%--
2      Created by IntelliJ IDEA.
3      User: rXing
4      Date: 4/14/2019
5      Time: 5:59 PM
6      To change this template use File | Settings | File Templates.
7  --%>
8  <%@ page contentType="text/html; charset=UTF-8" language="java" %>
9  <html>
10     <head>
11         <title>index</title>
12     </head>
13     <body>
14         Welcome to CS472 Web Programming
15     </body>
16 </html>
```

The code at this step is ready.

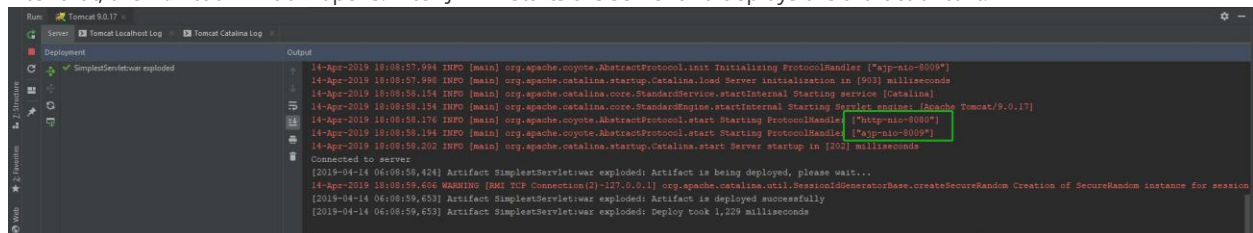
Running the application

In the upper-right part of the workspace, click .

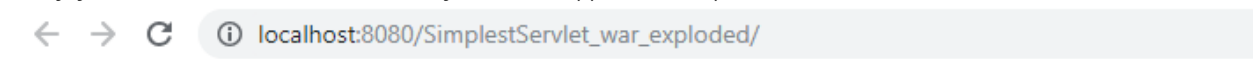


IntelliJ IDEA compiles your source code and builds an application [artifact](#).

After that, the **Run** tool window opens. IntelliJ IDEA starts the server and deploys the artifact onto it.



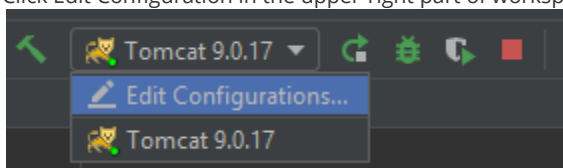
Finally, your default web browser starts and you see the application output **Hello, World!** there.



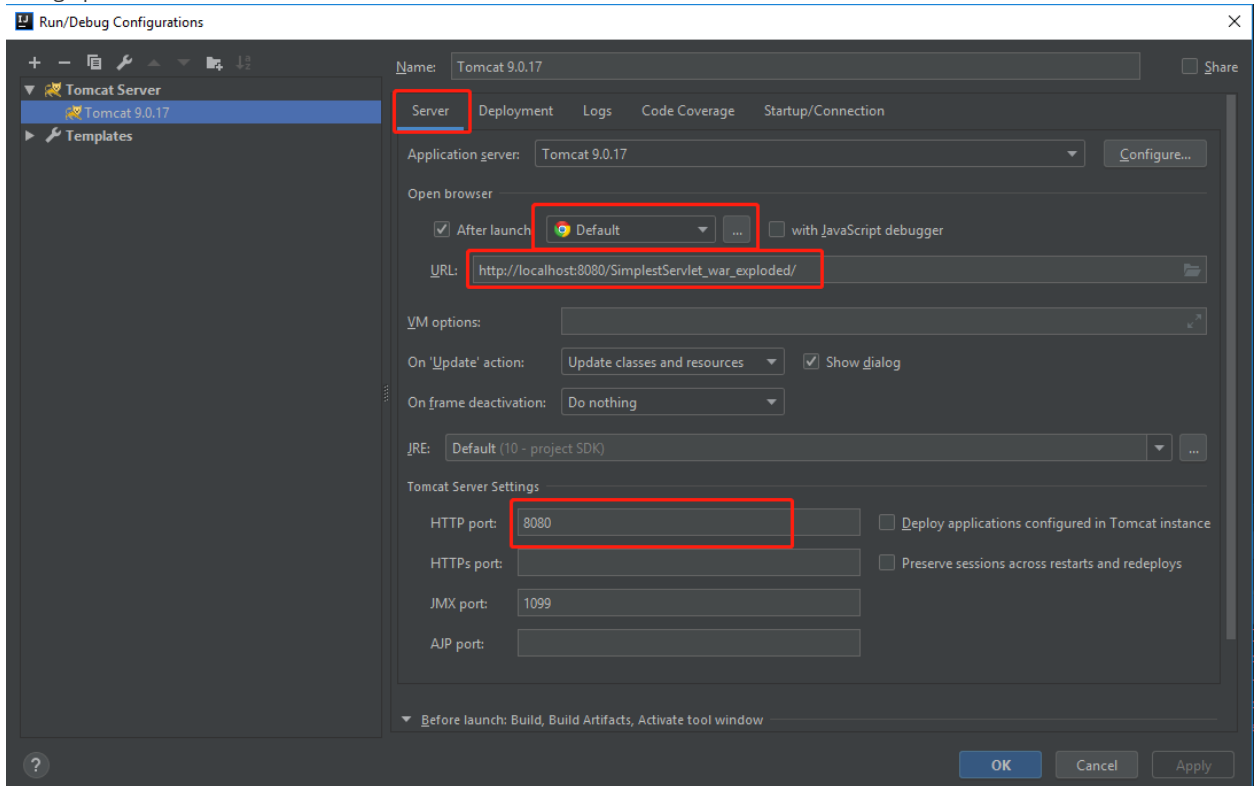
Welcome to CS472 Web Programming

Edit Configuration For Tomcat

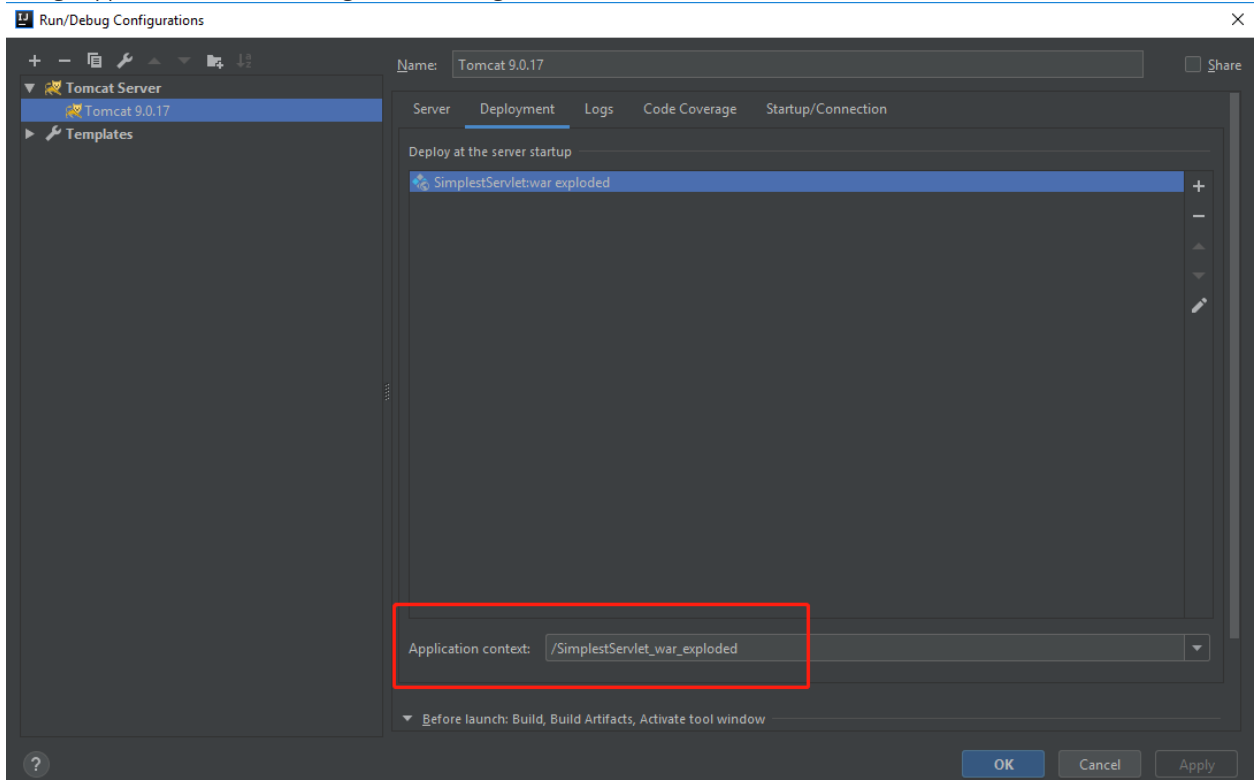
- Click Edit Configuration in the upper-right part of workspace



2. Change port number



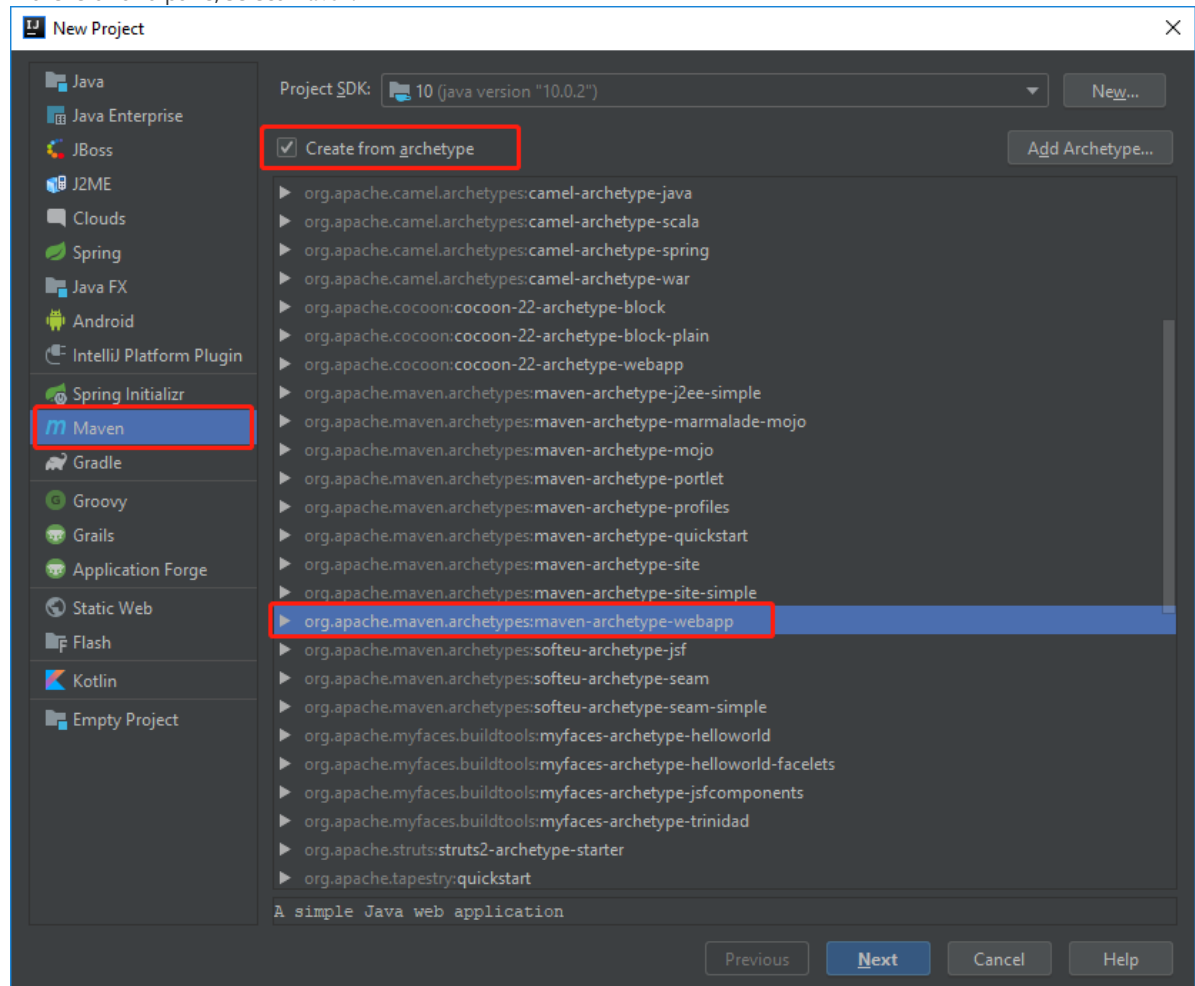
3. Change application context – change the accessing URL



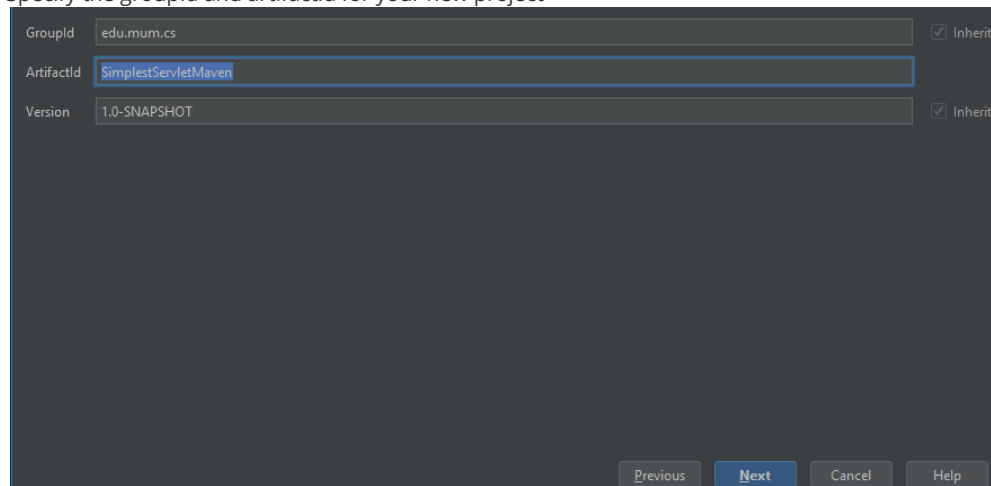
Create Maven Project

Creating a project

1. Click **Create New Project** on the Welcome screen, or select **File | New | Project**. The **New Project** wizard opens.
2. In the left-hand pane, select **Maven**.

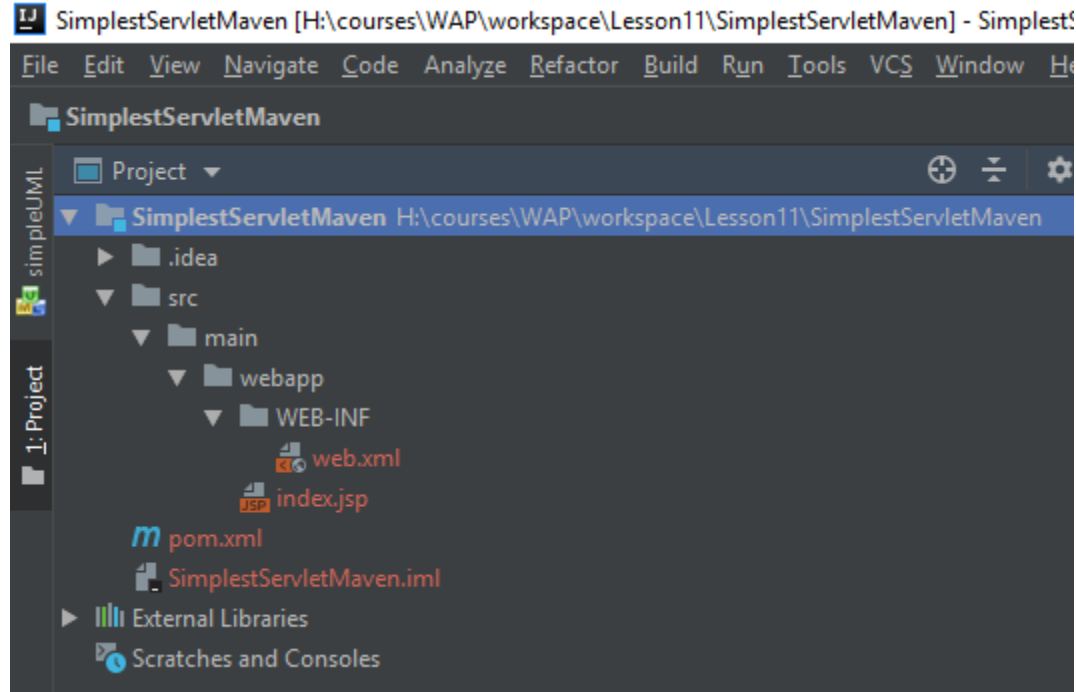


3. Specify the groupId and artifactId for your new project



Exploring the project structure

When the project is created, you'll see something similar to this in the **Project** tool window.



- **SimplestServletMaven** is a module folder (which in this case coincides with the project folder). The **.idea** folder and the file **SimplestServletMaven.iml** contain configuration data for your project and module respectively. The folder **src** is for your Java source code. The folder **web** is for the web part of your application. At the moment this folder contains the deployment descriptor **WEB-INF/web.xml** and the file **index.jsp** intended as a starting page of your application.
- **External Libraries** include your JDK.

Developing source code

Our application will be a single JSP page application. Its only function will be to output the text *Hello, World!*

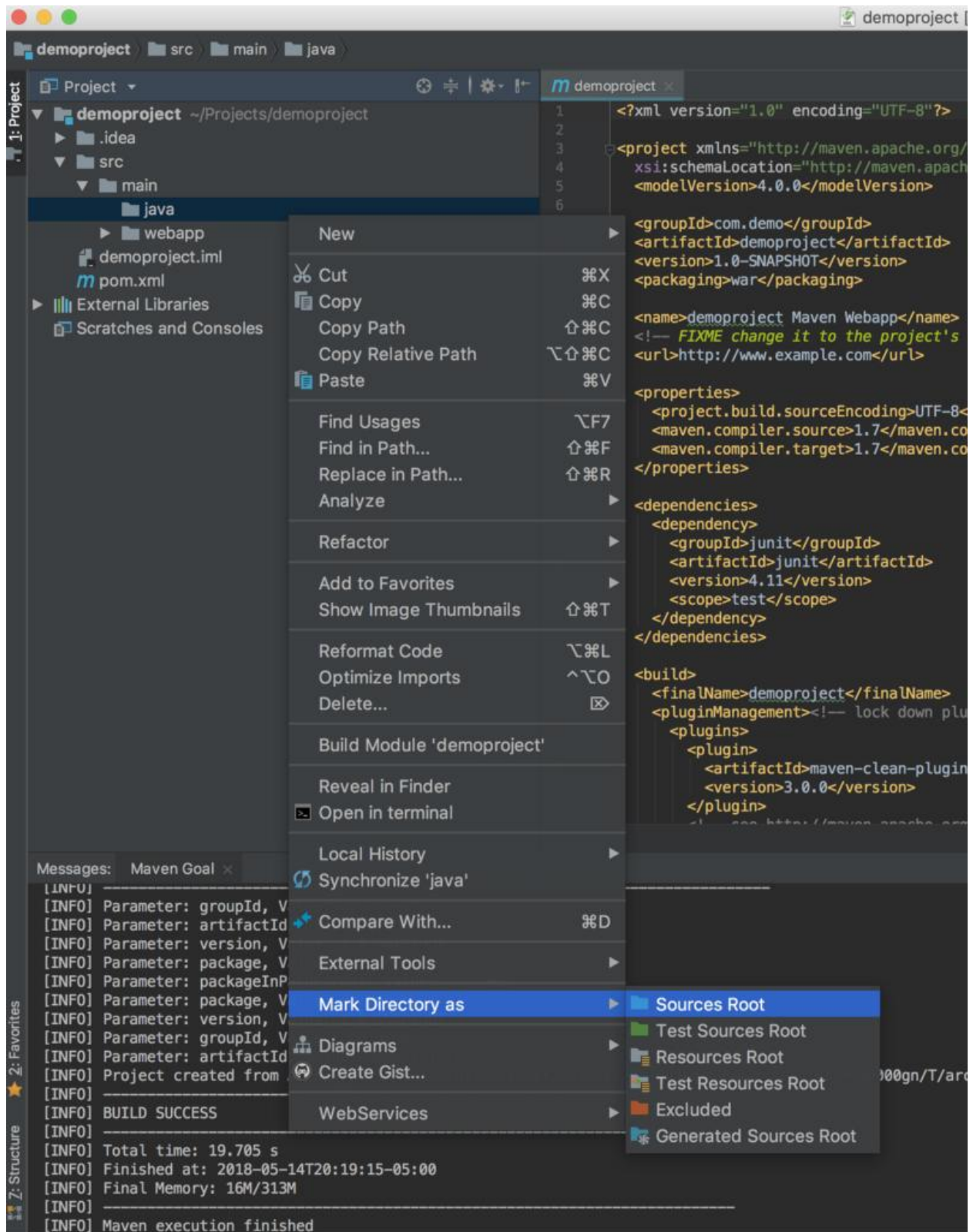
3. Add dependency in **pom.xml**

```
<dependency>
    <groupId>javax.servlet</groupId>
    <artifactId>javax.servlet-api</artifactId>
    <version>4.0.1</version>
    <scope>provided</scope>
</dependency>
```

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

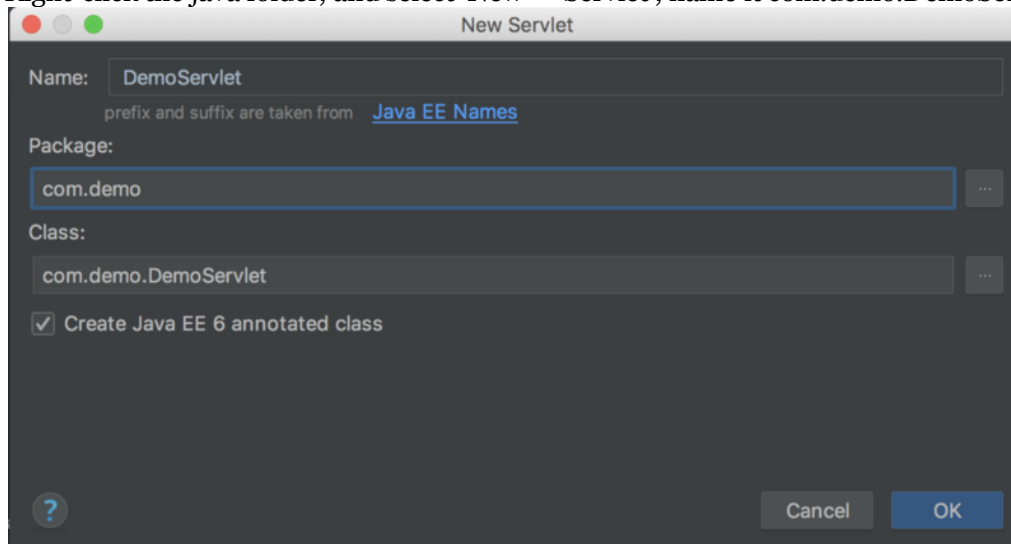
4. Create a servlet class

Firstly, create a new folder named **java** inside the **src.main** folder, then right-click the **java** folder, set it as sources root. If you don't do that, then the IntelliJ IDEA will not show 'Class' when we right-click and select 'New'.



Right-click the java folder, and select 'New' — 'Servlet', name it com.demo.DemoServlet.

Right-click the java folder, and select 'New' — 'Servlet', name it com.demo.DemoServlet.



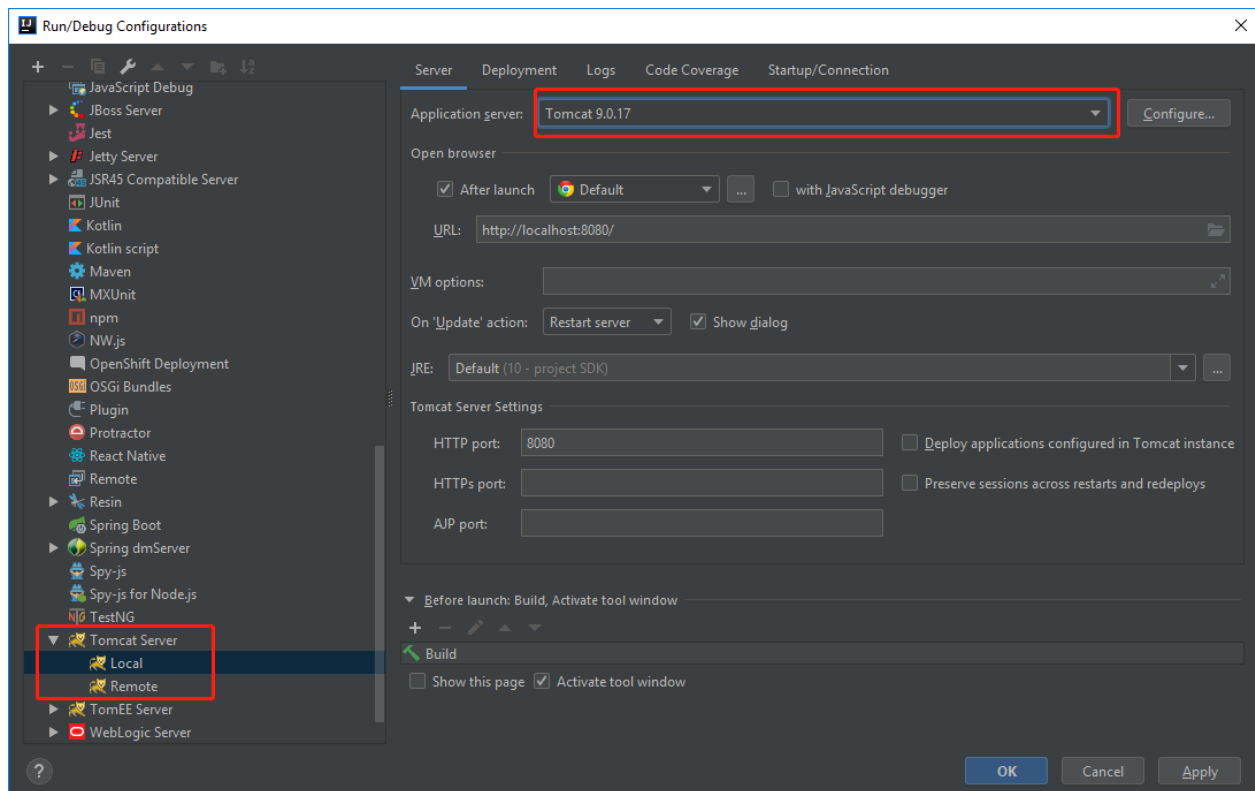
```
@WebServlet("/DemoServlet")
public class DemoServlet extends HttpServlet {
    protected void doPost(HttpServletRequest request, HttpServletResponse
response) throws ServletException, IOException {

    }

    protected void doGet(HttpServletRequest request, HttpServletResponse
response) throws ServletException, IOException {
        response.setContentType("text/html");
        PrintWriter out = response.getWriter();
        out.println("<h3>Hello World!</h3>");
    }
}
```

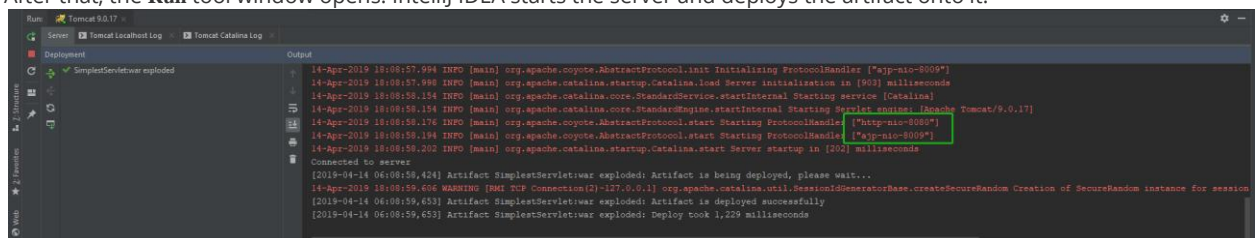
Running the application

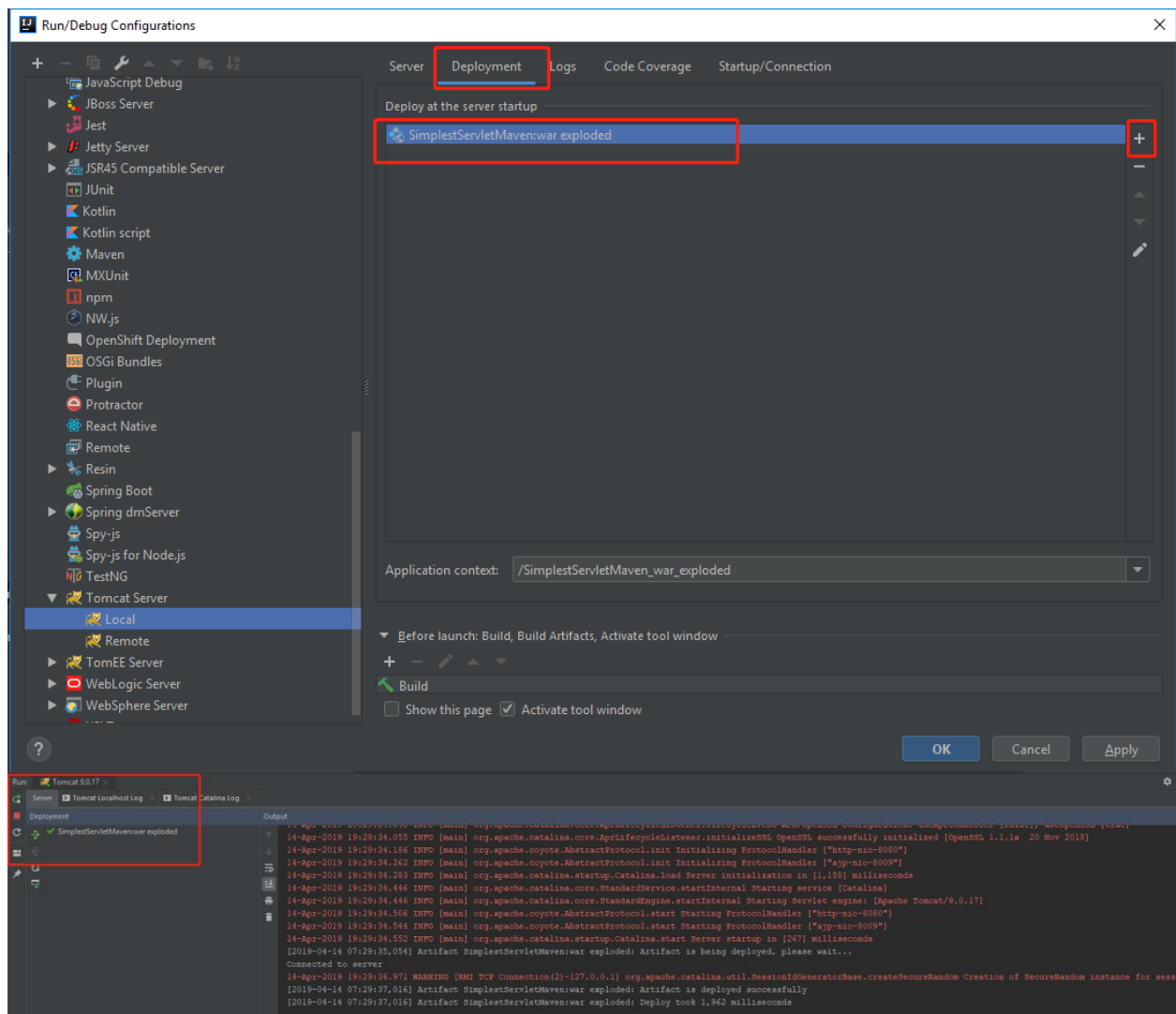
In the upper-right part of the workspace, click ►. Add Tomcat



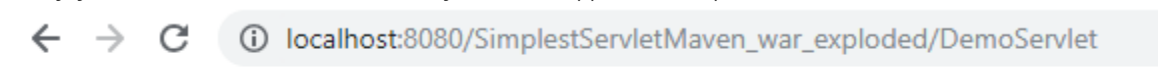
IntelliJ IDEA compiles your source code and builds an application [artifact](#).

After that, the **Run** tool window opens. IntelliJ IDEA starts the server and deploys the artifact onto it.





Finally, your default web browser starts and you see the application output **Hello, World!** there.



Hello World!

Credits:

<https://www.jetbrains.com/help/idea/creating-and-running-your-first-java-ee-application.html>

<https://medium.com/@backslash112/create-maven-project-with-servlet-in-intellij-idea-2018-be0d673bd9af>