

Software Engineering

LESSON 2

Servlet Technology

Outline

1. Introduction
2. Servlet API
3. Servlet Example





Overview

In this chapter, you are going to learn about

- Know Servlet
- Know how to create webpage using Servlet Technology
- Know how to use Servlet
- Know how to use Servlet
- Know how to implement website using Servlet



Learning content

1. Introduction

- What is Servlet?
- Flow of Servlet
- Web terminology

2. Servlet API

- Interfaces and classes in javax.servlet package
- Interfaces and classes in javax.servlet.http package
- Servlet interface

3. Servlet example

- Apache Tomcat
- Web module
- Servlet with Eclipse



1. Introduction

- **Servlet** technology is used to create a web application (resides at **server side** and generates a **dynamic web page**).
- **Servlet** technology is robust and scalable because of java language. Before Servlet, CGI (Common Gateway Interface) scripting language was common as a server-side programming language. However, there were many disadvantages to this technology. We have discussed these disadvantages below.
- There are many interfaces and classes in the Servlet API such as Servlet, GenericServlet, HttpServlet, ServletRequest, ServletResponse, etc.

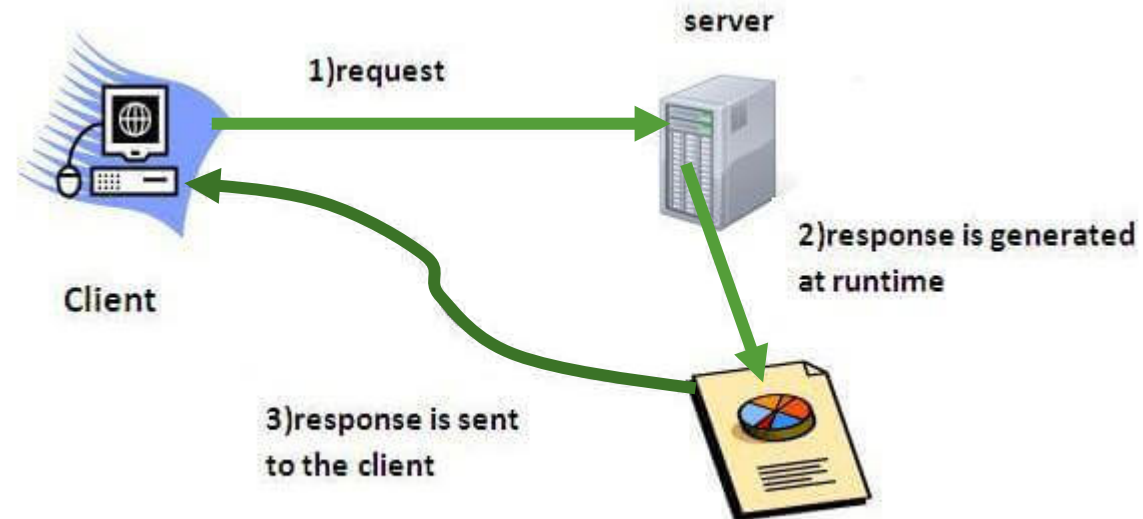


1.1. What is Servlet?

- Servlet is a technology used to create a web application.
- Servlet is an API that provides many interfaces and classes including documentation.
- Servlet is an interface that must be implemented for creating any Servlet.
- Servlet is a class that extends the capabilities of the servers and responds to the incoming requests. It can respond to any requests.
- Servlet is a web component that is deployed on the server to create a dynamic web page.

1.2. Flow of Servlet

<http://example.com:8080/>



1.3. Web terminology

Servlet Terminology	Description
<u>Website: static vs dynamic</u>	It is a collection of related web pages that may contain text, images, audio and video.
<u>HTTP</u>	It is the data communication protocol used to establish communication between client and server.
<u>HTTP Requests</u>	It is the request send by the computer to a web server that contains all sorts of potentially interesting information.
<u>Get vs Post</u>	It gives the difference between GET and POST request.
<u>Container</u>	It is used in java for dynamically generating the web pages on the server side.
<u>Server: Web vs Application</u>	It is used to manage the network resources and for running the program or software that provides services.
<u>Content Type</u>	It is HTTP header that provides the description about what are you sending to the browser.



2. Servlet API

- The **javax.servlet** package contains many interfaces and classes that are used by the servlet or web container. These are not specific to any protocol.
- The **javax.servlet.http** package contains interfaces and classes that are responsible for http requests only.

2.1. Interfaces and classes in javax.servlet package

Interfaces

- Servlet
- ServletRequest
- ServletResponse
- RequestDispatcher
- ServletConfig
- ServletContext
- SingleThreadModel
- Filter
- FilterConfig
- FilterChain
- ServletRequestListener
- ServletRequestAttributeListener
- ServletContextListener
- ServletContextAttributeListener

Classes

- GenericServlet
- ServletInputStream
- ServletOutputStream
- ServletRequestWrapper
- ServletResponseWrapper
- ServletRequestEvent
- ServletContextEvent
- ServletRequestAttributeEvent
- ServletContextAttributeEvent
- ServletException
- UnavailableException

2.2. Classes and Interfaces in javax.servlet.http package

Interfaces

- HttpServletRequest
- HttpServletResponse
- HttpSession
- HttpSessionListener
- HttpSessionAttributeListener
- HttpSessionBindingListener
- HttpSessionActivationListener
- HttpSessionContext (deprecated now)

Classes

- HttpServlet
- Cookie
- HttpServletRequestWrapper
- HttpServletResponseWrapper
- HttpSessionEvent
- HttpSessionBindingEvent
- HttpUtils (deprecated now)



2.3. Servlet Interface

- **Servlet interface provides** common behavior to all the servlets. Servlet interface defines **methods that all servlets must implement**.
- Servlet interface needs to be implemented for creating any servlet (either directly or indirectly). It provides **3 life cycle methods** that are used to **initialize** the servlet, to **service** the requests, and to **destroy** the servlet and 2 non-life cycle methods.

2.3. Servlet Interface

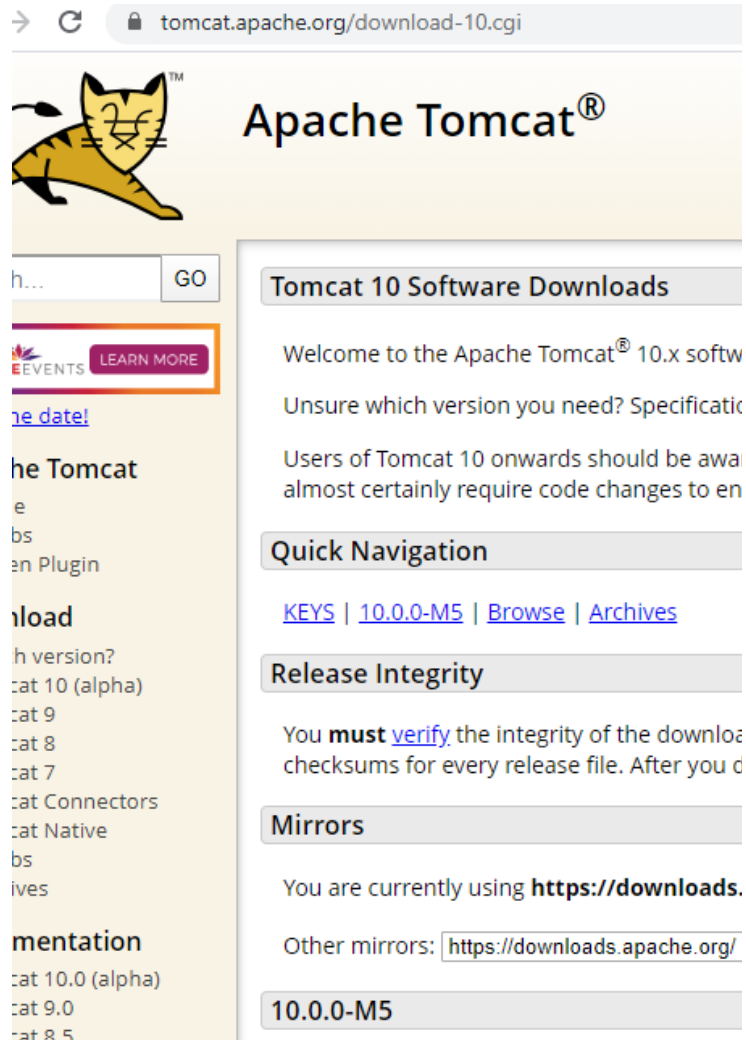
Method	Description
public void init(ServletConfig config)	initializes the servlet. It is the life cycle method of servlet and invoked by the web container only once.
public void service(ServletRequest request, ServletResponse response)	provides response for the incoming request. It is invoked at each request by the web container.
public void destroy()	is invoked only once and indicates that servlet is being destroyed.
public ServletConfig getServletConfig()	returns the object of ServletConfig.
public String getServletInfo()	returns information about servlet such as writer, copyright, version etc.

3. Servlet Example

- Prerequisite
 - Apache Tomcat (Web server Java like Apache PHP)
 - OpenJDK or JDK 8.0+
 - Eclipse IDE for JEE developer



3.1. Apache Tomcat



- Download Apache Tomcat 9.x at: <https://tomcat.apache.org/download-90.cgi>
- Download as **64-bit Windows zip**
- After download, extract it, and then a folder is extracted with name: apache-tomcat-9.0.35 or any version number greater than 8.



3.1. Apache Tomcat

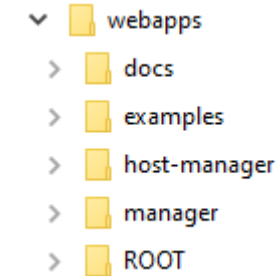
- bin
- conf
- lib
- logs
- temp
- webapps
- work

- **/bin** - Startup, shutdown, and other scripts. The *.sh files (for Unix systems) are functional duplicates of the *.bat files (for Windows systems). Since the Win32 command-line lacks certain functionality, there are some additional files in here.
- **/conf** - Configuration files and related DTDs. The most important file in here is server.xml. It is the main configuration file for the container.
- **/logs** - Log files are here by default.
- **/webapps** - This is where your webapps go.



3.1. Apache Tomcat

- **docs/** - user manual for Tomcat users
- **examples/** - examples web apps
- **host-manager/** - Web App enables you to create, delete, and manage virtual hosts within Tomcat
- **manager/** - a web app used to deploy a new web application, or undeploy an existing one, without having to shut down and restart the entire container
- **ROOT/** - root web app (ex: <http://localhost:8080/>)





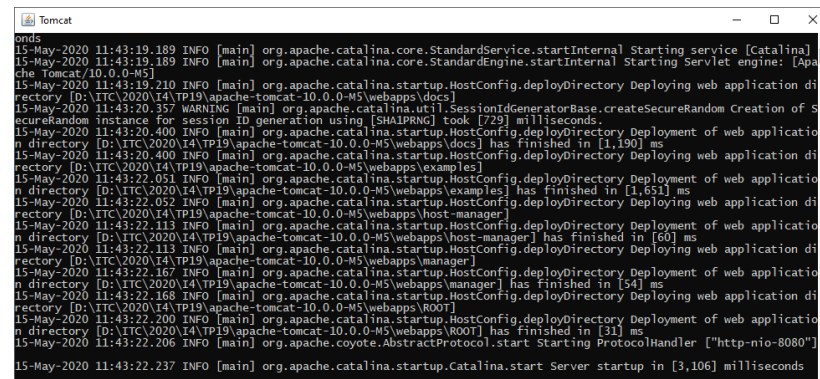
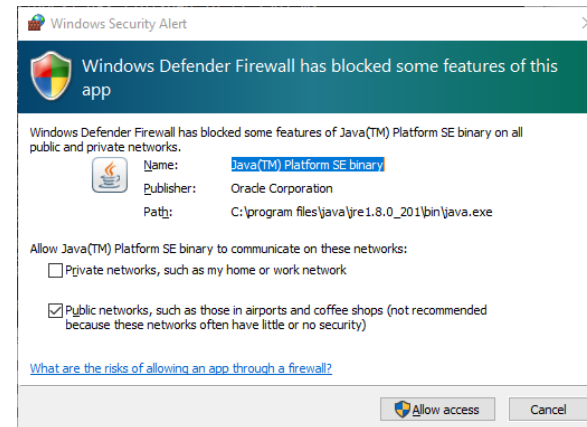
3.1. Apache Tomcat

• Starting Server:

- Double click on bin/startup.bat
- If prompt, click allow:

- If successful startup, it will show interface like bellow:
- **DO NOT CLOSE THIS DOS WINDOWS UNLESS SHUTDOWN SERVER!**

- ▼ webapps
 - > docs
 - > examples
 - > host-manager
 - > manager
 - > ROOT



3.1. Apache Tomcat

- Access to server with address: <http://localhost:8080/>
- Or any port indicated in the DOS windows when start up successfully.


```
Select Tomcat
15-May-2020 11:43:17.937 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log Server version name: Apache Tomcat/10.0.0-M5
15-May-2020 11:43:17.966 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log Server built: May 5 2020 18:38:15 UTC
15-May-2020 11:43:17.971 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log Server version number: 10.0.0.0
15-May-2020 11:43:17.971 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log OS Name: Windows 10
15-May-2020 11:43:17.971 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log OS Version: 10.0
15-May-2020 11:43:17.972 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log Architecture: amd64
15-May-2020 11:43:17.972 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log Java Home: C:\Program Files\Java\jre1.8.0_201
15-May-2020 11:43:17.972 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log JVM Version: 1.8.0_201-b09
15-May-2020 11:43:17.972 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log JVM Vendor: Oracle Corporation
15-May-2020 11:43:17.973 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log CATALINA_BASE: D:\ITC\2020\I4\TP19\apache-tomcat-10.0.0-M5
15-May-2020 11:43:17.973 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log CATALINA_HOME: D:\ITC\2020\I4\TP19\apache-tomcat-10.0.0-M5
15-May-2020 11:43:17.973 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log Command line argument: -Djava.util.logging.config.file=D:\ITC\2020\I4\TP19\apache-tomcat-10.0.0-M5\conf\logging.properties
15-May-2020 11:43:17.974 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log Command line argument: -Djava.util.logging.manager=org.apache.juli.ClassLoaderLogManager
15-May-2020 11:43:17.974 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log Command line argument: -Djdk.tls.ephemeralDHKeySize=2048
15-May-2020 11:43:17.974 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log Command line argument: -Djava.protocol.handler.pkgs=org.apache.catalina.webresources
15-May-2020 11:43:17.974 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log Command line argument: -Dignore.endorsed.dirs=
15-May-2020 11:43:17.975 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log Command line argument: -Dcatalina.base=D:\ITC\2020\I4\TP19\apache-tomcat-10.0.0-M5
15-May-2020 11:43:17.975 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log Command line argument: -Dcatalina.home=D:\ITC\2020\I4\TP19\apache-tomcat-10.0.0-M5
15-May-2020 11:43:17.975 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log Command line argument: -Djava.io.tmpdir=D:\ITC\2020\I4\TP19\apache-tomcat-10.0.0-M5\temp
15-May-2020 11:43:17.977 INFO [main] org.apache.catalina.startup.VersionLoggerListener.log
15-May-2020 11:43:18.005 INFO [main] org.apache.catalina.core.AprLifecycleListener.lifecycleEvent Loaded Apache Tomcat Native library [1.2.24] using APR version [1.7.0].
15-May-2020 11:43:18.005 INFO [main] org.apache.catalina.core.AprLifecycleListener.lifecycleEvent APR capabilities: IPV6 [true], sendfile [true], accept filters [false], random [true].
15-May-2020 11:43:18.016 INFO [main] org.apache.catalina.core.AprLifecycleListener.initializeSSL OpenSSL successfully initialized [OpenSSL 1.1.1g 21 Apr 2020]
15-May-2020 11:43:19.079 INFO [main] org.apache.coyote.AbstractProtocol.init Initializing ProtocolHandler ["http-nio-8080"]
15-May-2020 11:43:19.129 INFO [main] org.apache.catalina.startup.Catalina.load Server initialization in [1,491] milliseconds
15-May-2020 11:43:19.189 INFO [main] org.apache.catalina.core.StandardService.startInternal Starting service [Catalina]
15-May-2020 11:43:19.189 INFO [main] org.apache.catalina.core.StandardEngine.startInternal Starting Servlet engine: [Apache Tomcat/10.0.0-M5]
15-May-2020 11:43:19.210 INFO [main] org.apache.catalina.startup.HostConfig.deployDirectory Deploying web application directory [D:\ITC\2020\I4\TP19\apache-tomcat-10.0.0-M5\webapps\docs]
15-May-2020 11:43:20.357 WARNING [main] org.apache.catalina.util.SessionIdGeneratorBase.createSecureRandom Creation of SecureRandom instance for session ID generation using [SHA1PRNG] took [729] milliseconds.
15-May-2020 11:43:20.400 INFO [main] org.apache.catalina.startup.HostConfig.deployDirectory Deployment of web application directory [D:\ITC\2020\I4\TP19\apache-tomcat-10.0.0-M5\webapps\docs] has finished in [1,190] ms
15-May-2020 11:43:20.400 INFO [main] org.apache.catalina.startup.HostConfig.deployDirectory Deploying web application directory [D:\ITC\2020\I4\TP19\apache-tomcat-10.0.0-M5\webapps\examples]
15-May-2020 11:43:22.051 INFO [main] org.apache.catalina.startup.HostConfig.deployDirectory Deployment of web application directory [D:\ITC\2020\I4\TP19\apache-tomcat-10.0.0-M5\webapps\examples] has finished in [1,651] ms
15-May-2020 11:43:22.052 INFO [main] org.apache.catalina.startup.HostConfig.deployDirectory Deploying web application directory [D:\ITC\2020\I4\TP19\apache-tomcat-10.0.0-M5\webapps\host-manager]
15-May-2020 11:43:22.113 INFO [main] org.apache.catalina.startup.HostConfig.deployDirectory Deployment of web application directory [D:\ITC\2020\I4\TP19\apache-tomcat-10.0.0-M5\webapps\host-manager] has finished in [60] ms
15-May-2020 11:43:22.113 INFO [main] org.apache.catalina.startup.HostConfig.deployDirectory Deploying web application directory [D:\ITC\2020\I4\TP19\apache-tomcat-10.0.0-M5\webapps\manager]
15-May-2020 11:43:22.167 INFO [main] org.apache.catalina.startup.HostConfig.deployDirectory Deployment of web application directory [D:\ITC\2020\I4\TP19\apache-tomcat-10.0.0-M5\webapps\manager] has finished in [54] ms
15-May-2020 11:43:22.168 INFO [main] org.apache.catalina.startup.HostConfig.deployDirectory Deploying web application directory [D:\ITC\2020\I4\TP19\apache-tomcat-10.0.0-M5\webapps\ROOT]
15-May-2020 11:43:22.200 INFO [main] org.apache.catalina.startup.HostConfig.deployDirectory Deployment of web application directory [D:\ITC\2020\I4\TP19\apache-tomcat-10.0.0-M5\webapps\ROOT] has finished in [31] ms
15-May-2020 11:43:22.206 INFO [main] org.apache.coyote.AbstractProtocol.start Starting ProtocolHandler ["http-nio-8080"]
15-May-2020 11:43:22.237 INFO [main] org.apache.catalina.startup.Catalina.start Server startup in [3,106] milliseconds
```




3.1. Apache Tomcat

- If success start up, web page at address:
<http://localhost:8080/>

[Home](#) [Documentation](#) [Configuration](#) [Examples](#) [Wiki](#) [Mailing Lists](#) [Find Help](#)

Apache Tomcat/10.0.0-M5 

If you're seeing this, you've successfully installed Tomcat. Congratulations!



Recommended Reading:
[Security Considerations How-To](#)
[Manager Application How-To](#)
[Clustering/Session Replication How-To](#)

[Server Status](#)
[Manager App](#)
[Host Manager](#)

Developer Quick Start
[Tomcat Setup](#) [Realms & AAA](#) [Examples](#) [Servlet Specifications](#)
[First Web Application](#) [JDBC Data Sources](#) [Tomcat Versions](#)

Managing Tomcat

For security, access to the [manager webapp](#) is restricted. Users are defined in:

```
$CATALINA_HOME/conf/tomcat-users.xml
```

In Tomcat 10.0 access to the manager application is split between different users.
[Read more...](#)

[Release Notes](#)
[Changelog](#)
[Migration Guide](#)
[Security Notices](#)

Documentation

[Tomcat 10.0 Documentation](#)
[Tomcat 10.0 Configuration](#)
[Tomcat Wiki](#)

Find additional important configuration information in:

```
$CATALINA_HOME/RUNNING.txt
```

Developers may be interested in:

[Tomcat 10.0 Bug Database](#)
[Tomcat 10.0 JavaDocs](#)
[Tomcat 10.0 Git Repository at GitHub](#)

Getting Help

FAQ and Mailing Lists

The following mailing lists are available:

[tomcat-announce](#)
Important announcements, releases, security vulnerability notifications. (Low volume).

[tomcat-users](#)
User support and discussion

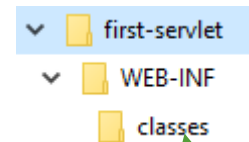
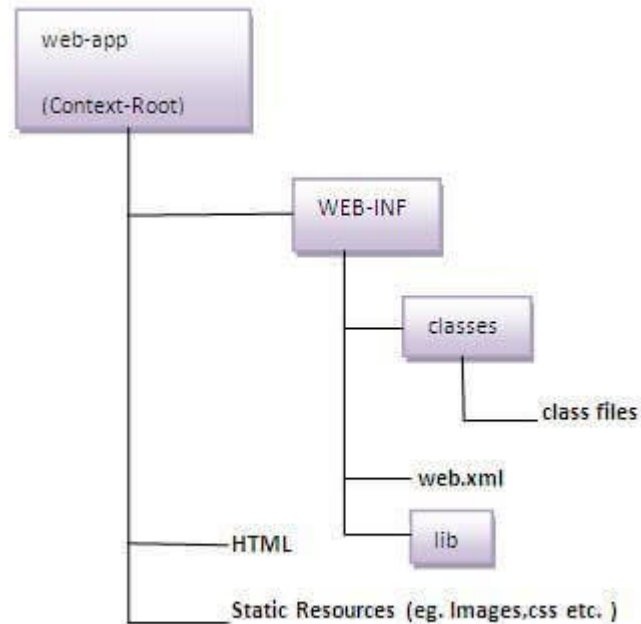
[taglibs-user](#)
User support and discussion for [Apache Taglibs](#)

[tomcat-dev](#)
Development mailing list, including commit messages

[Other Downloads](#) [Other Documentation](#) [Get Involved](#) [Miscellaneous](#) [Apache Software](#)

3.2. Web module

- Folder structure of a Web App in Tomcat:



Your servlet classes here.

3.2. Web module

- First.java file created in folder first-servlet/WEB-INF/classes, then Enter content:

```
import java.io.*;

import javax.servlet.*;

public class First implements Servlet {
    ServletConfig config = null;

    public void init(ServletConfig config) {
        this.config = config;
        System.out.println("servlet is initialized");
    }

    public void service(ServletRequest req, ServletResponse res)
        throws IOException, ServletException {

        res.setContentType("text/html");

        PrintWriter out = res.getWriter();
        out.print("<html><body>");
        out.print("<b>hello simple servlet</b>");
        out.print("</body></html>");
    }

    public void destroy() {
        System.out.println("servlet is destroyed");
    }

    public ServletConfig getServletConfig() {
        return config;
    }

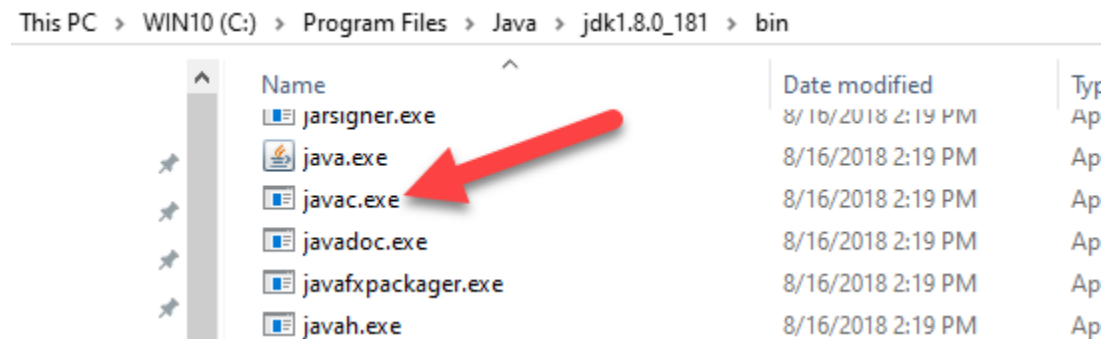
    public String getServletInfo() {
        return "copyright 2007-1010";
    }
}
```

3.2. Web module

- Save the file First.java and then compile:

```
javac -cp ../../../../../../lib/servlet-api.jar First.java
```

- The javac.exe is located in <JDK_DIR>/bin (example: C:\Program Files\Java\jdk1.8.0_181\bin\javac.exe)



3.2. Web module

- If Javac.exe is not found, it will message as bellow:

```
Microsoft Windows [Version 10.0.18362.836]
(c) 2019 Microsoft Corporation. All rights reserved.

C:\Users\DICE>cd D:\ITC\2020\I4\TP19\apache-tomcat-9.0.35\webapps\first-jsp\WEB-INF\classes

C:\Users\DICE>D:

D:\ITC\2020\I4\TP19\apache-tomcat-9.0.35\webapps\first-jsp\WEB-INF\classes>javac First.java
'javac' is not recognized as an internal or external command,
operable program or batch file.

D:\ITC\2020\I4\TP19\apache-tomcat-9.0.35\webapps\first-jsp\WEB-INF\classes>_
```

- In this case you need to put full path

```
D:\ITC\2020\I4\TP19\apache-tomcat-9.0.35\webapps\first-jsp\WEB-
INF\classes>"C:\Program Files\Java\jdk1.8.0_181\bin\javac.exe" -cp
../../../../lib/servlet-api.jar First.java

D:\ITC\2020\I4\TP19\apache-tomcat-9.0.35\webapps\first-jsp\WEB-INF\classes>
```




3.2. Web module

- Analysis

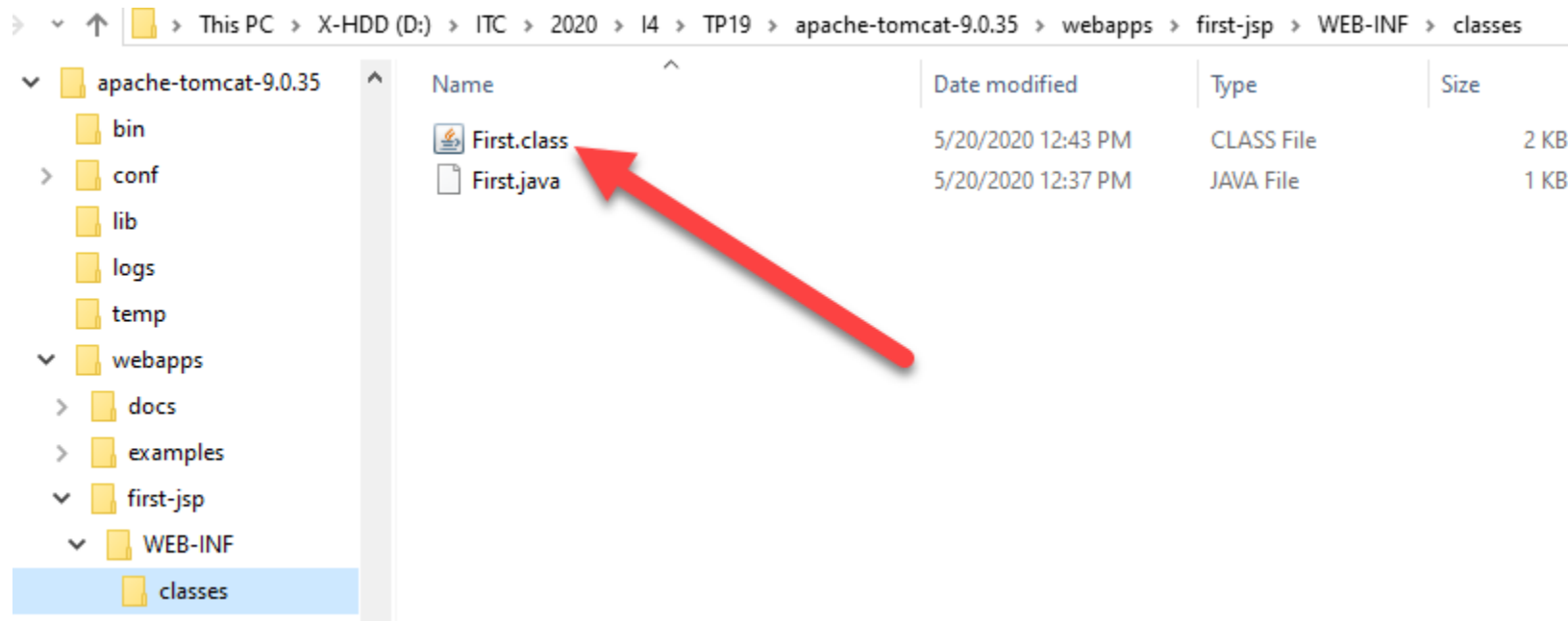
```
D:\ITC\2020\I4\TP19\apache-tomcat-9.0.35\webapps\first-jsp\WEB-INF\classes>"C:\Program Files\Java\jdk1.8.0_181\bin\javac.exe" -cp  
../../../../lib/servlet-api.jar First.java
```

```
D:\ITC\2020\I4\TP19\apache-tomcat-9.0.35\webapps\first-jsp\WEB-INF\classes>
```

- -cp: means to use Library (*.jar) (CLASSPATH)
- -cp ../../../../lib/servlet-api.jar
Means that we need Servlet-API from Tomcat/lib/servlet-api.jar

3.2. Web module

- If compile successful, it will generate a class file (*.class) in WEB-INF/classes:



3.2. Web module

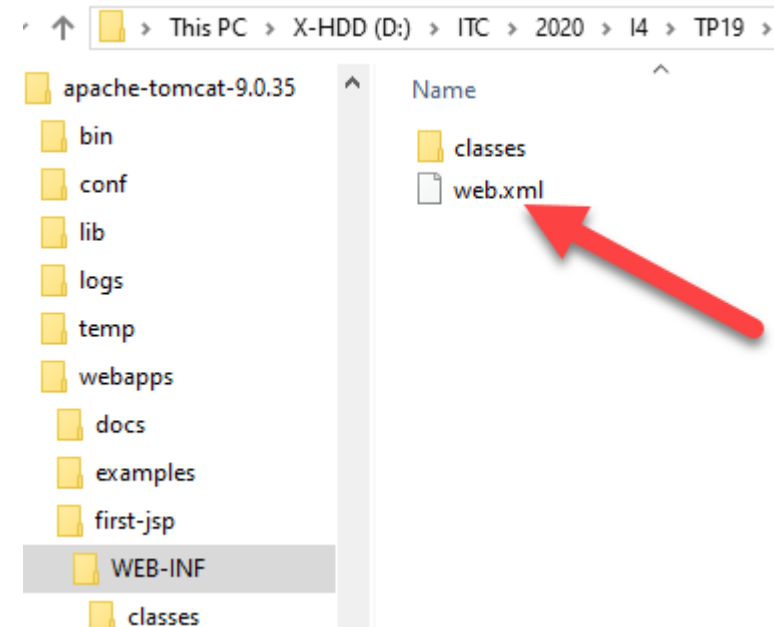
- Next, we create web.xml file in /WEB-INF/ with content:

```
<web-app>

  <servlet>
    <servlet-name>my-first-servlet</servlet-name>
    <servlet-class>First</servlet-class>
  </servlet>

  <servlet-mapping>
    <servlet-name>my-first-servlet</servlet-name>
    <url-pattern>/welcome</url-pattern>
  </servlet-mapping>

</web-app>
```



3.2. Web module

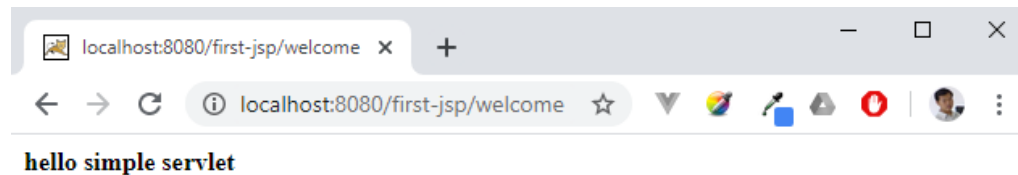
- Now, it is all setup!!
- You can start (or restart the server) to apply the changes
 - To do that, you can close the server console, and then run startup.bat again
 - After restarted the server, you will notice the different:

```
Tomcat
15-May-2020 11:43:19.189 INFO [main] org.apache.catalina.core.StandardService.startInternal Starting service [catalina]
15-May-2020 11:43:19.189 INFO [main] org.apache.catalina.core.StandardEngine.startInternal Starting Servlet engine: [Apache Tomcat/9.0.0-M5]
15-May-2020 11:43:19.210 INFO [main] org.apache.catalina.startup.HostConfig.deployDirectory Deploying web application directory [D:\ITC\2020\I4\TP19\apache-tomcat-9.0.0-M5\webapps\docs]
15-May-2020 11:43:20.357 WARNING [main] org.apache.catalina.util.SessionIdGeneratorBase.createSecureRandom Creation of SecureRandom instance for session ID generation using [SHA1PRNG] took [729] milliseconds
15-May-2020 11:43:20.400 INFO [main] org.apache.catalina.startup.HostConfig.deployDirectory Deployment of web application directory [D:\ITC\2020\I4\TP19\apache-tomcat-9.0.0-M5\webapps\docs] has finished in [1190] ms
15-May-2020 11:43:20.400 INFO [main] org.apache.catalina.startup.HostConfig.deployDirectory Deploying web application directory [D:\ITC\2020\I4\TP19\apache-tomcat-9.0.0-M5\webapps\examples]
15-May-2020 11:43:22.051 INFO [main] org.apache.catalina.startup.HostConfig.deployDirectory Deployment of web application directory [D:\ITC\2020\I4\TP19\apache-tomcat-9.0.0-M5\webapps\examples] has finished in [1161] ms
15-May-2020 11:43:22.052 INFO [main] org.apache.catalina.startup.HostConfig.deployDirectory Deploying web application directory [D:\ITC\2020\I4\TP19\apache-tomcat-9.0.0-M5\webapps\host-manager]
15-May-2020 11:43:22.113 INFO [main] org.apache.catalina.startup.HostConfig.deployDirectory Deployment of web application directory [D:\ITC\2020\I4\TP19\apache-tomcat-9.0.0-M5\webapps\host-manager] has finished in [60] ms
15-May-2020 11:43:22.113 INFO [main] org.apache.catalina.startup.HostConfig.deployDirectory Deploying web application directory [D:\ITC\2020\I4\TP19\apache-tomcat-9.0.0-M5\webapps\manager]
15-May-2020 11:43:22.167 INFO [main] org.apache.catalina.startup.HostConfig.deployDirectory Deployment of web application directory [D:\ITC\2020\I4\TP19\apache-tomcat-9.0.0-M5\webapps\manager] has finished in [54] ms
15-May-2020 11:43:22.168 INFO [main] org.apache.catalina.startup.HostConfig.deployDirectory Deploying web application directory [D:\ITC\2020\I4\TP19\apache-tomcat-9.0.0-M5\webapps\ROOT]
15-May-2020 11:43:22.200 INFO [main] org.apache.catalina.startup.HostConfig.deployDirectory Deployment of web application directory [D:\ITC\2020\I4\TP19\apache-tomcat-9.0.0-M5\webapps\ROOT] has finished in [31] ms
15-May-2020 11:43:22.206 INFO [main] org.apache.coyote.AbstractProtocol.start Starting ProtocolHandler ["http-nio-8080"]
15-May-2020 11:43:22.237 INFO [main] org.apache.catalina.startup.Catalina.start Server startup in [3,106] milliseconds
```

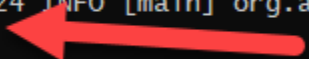
```
Starting Servlet engine: [Apache Tomcat/9.0.0-M5]
Deploying web application directory [D:\ITC\2020\I4\TP19\apache-tomcat-9.0.35\webapps\docs]
Deployment of web application directory [D:\ITC\2020\I4\TP19\apache-tomcat-9.0.35\webapps\docs] has finished in [347] ms
Deploying web application directory [D:\ITC\2020\I4\TP19\apache-tomcat-9.0.35\webapps\examples]
Deployment of web application directory [D:\ITC\2020\I4\TP19\apache-tomcat-9.0.35\webapps\examples] has finished in [370] ms
Deploying web application directory [D:\ITC\2020\I4\TP19\apache-tomcat-9.0.35\webapps\first-jsp]
Deployment of web application directory [D:\ITC\2020\I4\TP19\apache-tomcat-9.0.35\webapps\first-jsp] has finished in [45] ms
Deploying web application directory [D:\ITC\2020\I4\TP19\apache-tomcat-9.0.35\webapps\host-manager]
Deployment of web application directory [D:\ITC\2020\I4\TP19\apache-tomcat-9.0.35\webapps\host-manager] has finished in [40] ms
Deploying web application directory [D:\ITC\2020\I4\TP19\apache-tomcat-9.0.35\webapps\manager]
Deployment of web application directory [D:\ITC\2020\I4\TP19\apache-tomcat-9.0.35\webapps\manager] has finished in [42] ms
Deploying web application directory [D:\ITC\2020\I4\TP19\apache-tomcat-9.0.35\webapps\ROOT]
Deployment of web application directory [D:\ITC\2020\I4\TP19\apache-tomcat-9.0.35\webapps\ROOT] has finished in [27] ms
colHandler ["http-nio-8080"]
up in [959] milliseconds
```

3.2. Web module

- To test whether it works, open your browser, and enter the address below:
<http://localhost:8080/first-jsp/welcome>
- If everything's fine, you will see in browser below:



```
20-May-2020 12:58:37.924 INFO [main] org.apache.catalina.s  
servlet is initialized
```



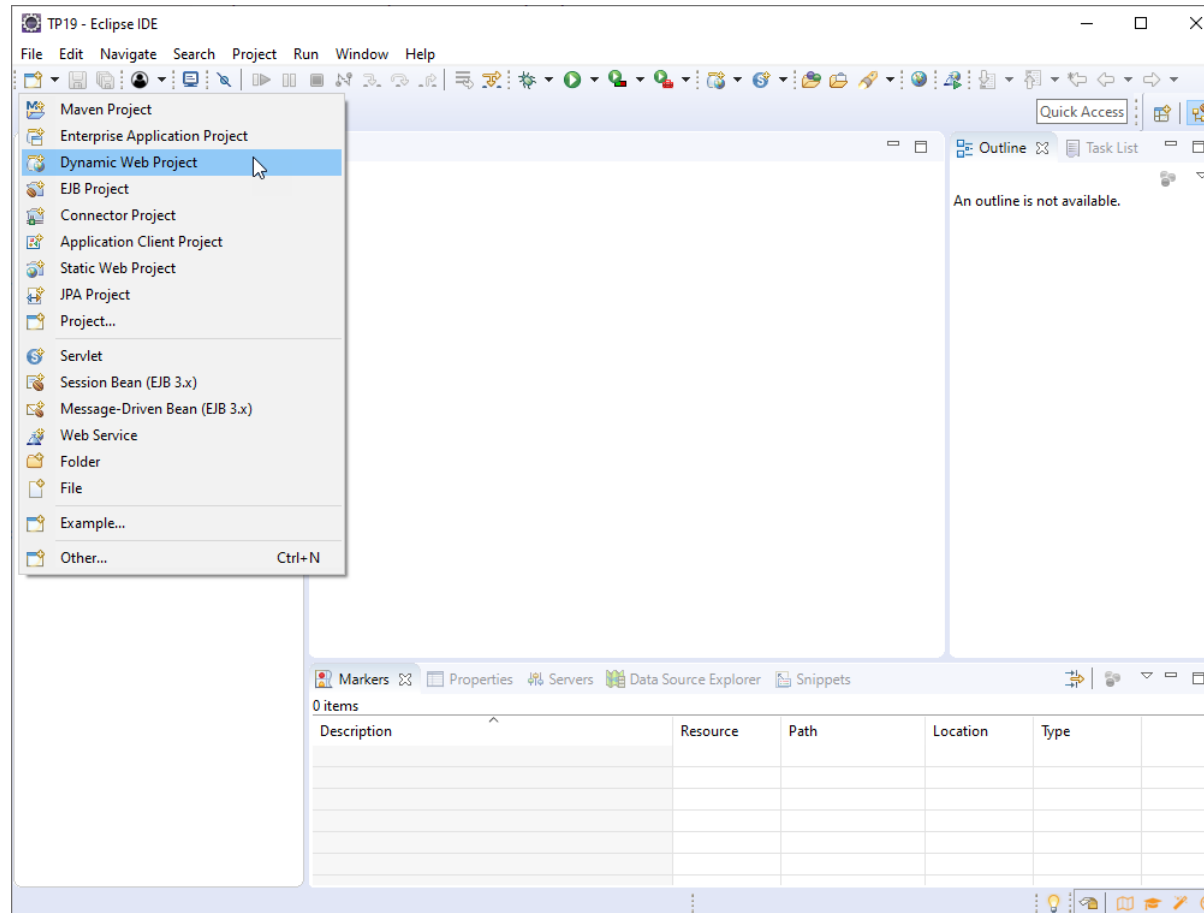


3.3. Servlet with Eclipse

- Eclipse is an open-source ide for developing JavaSE and JavaEE (J2EE) applications. You can download the eclipse ide from the eclipse website <http://www.eclipse.org/downloads/>.
- You need to download the eclipse ide for **JavaEE developers**.
- Steps:
 1. Create a Dynamic web project
 2. Create a servlet
 3. Run the servlet

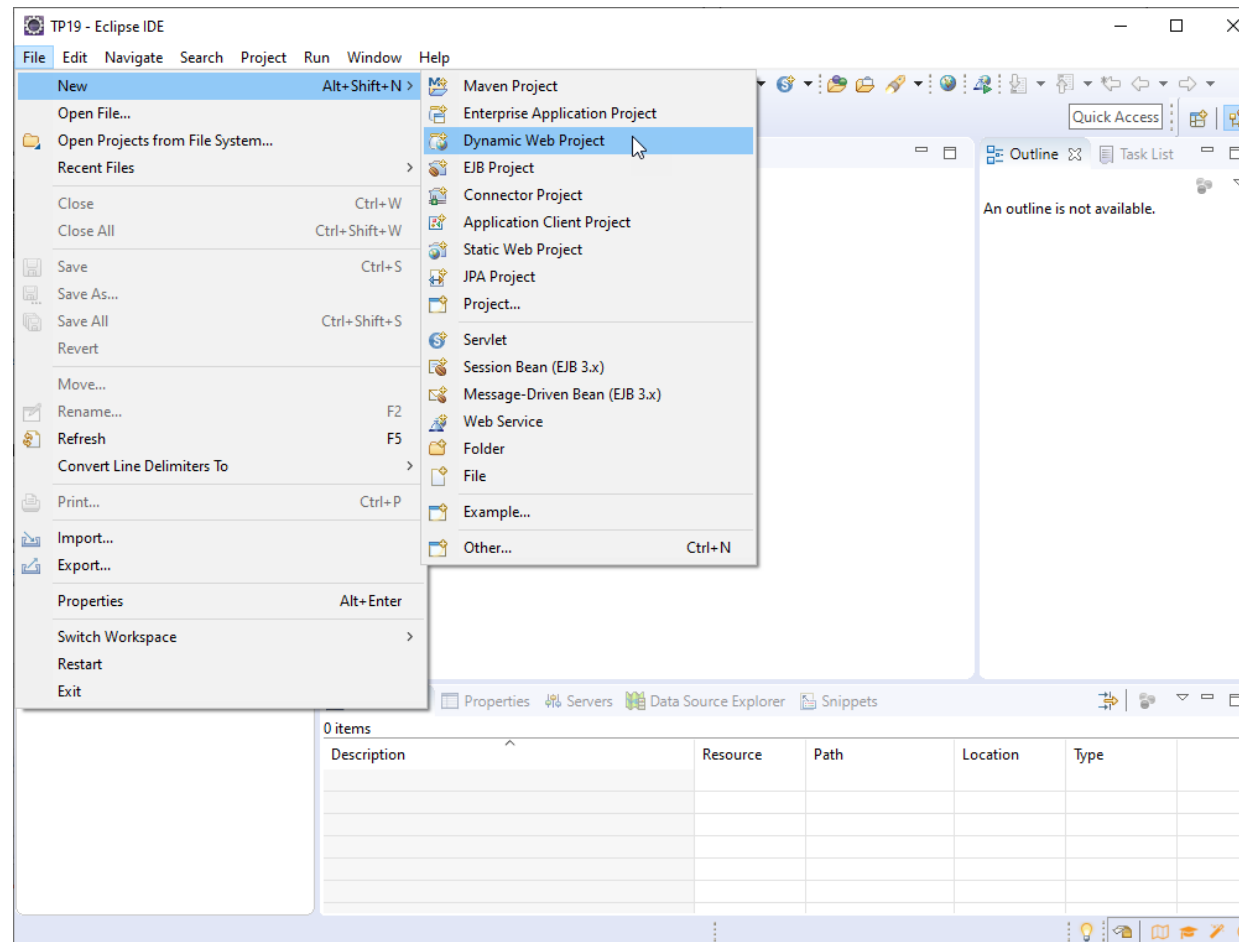
3.3. Servlet with Eclipse

1. Create a Dynamic web project (Method 1)



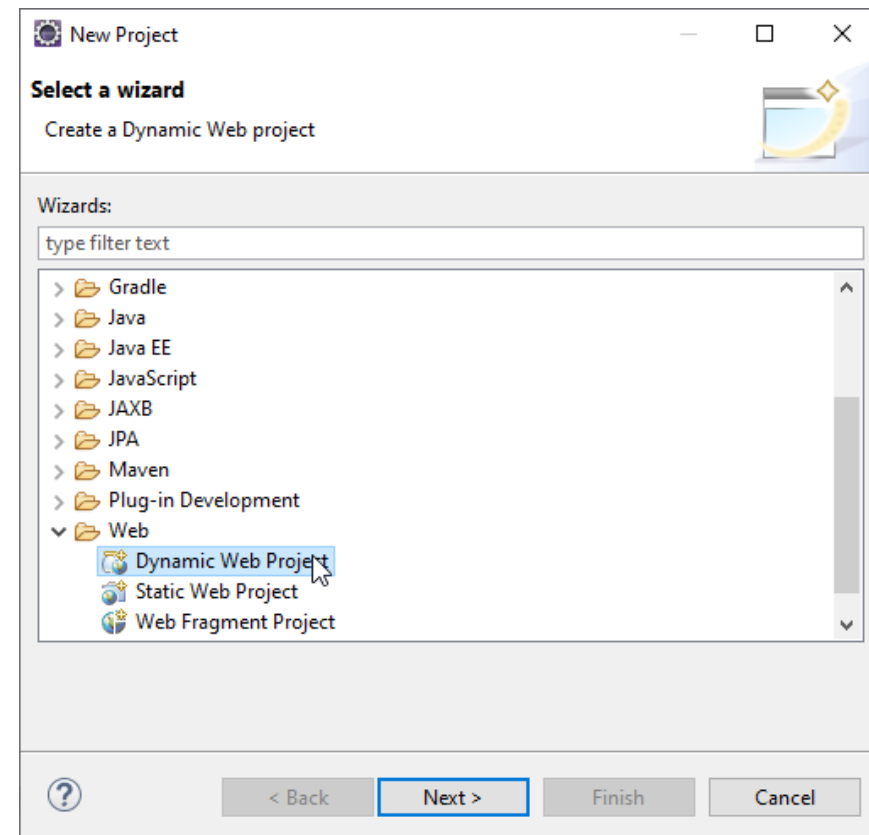
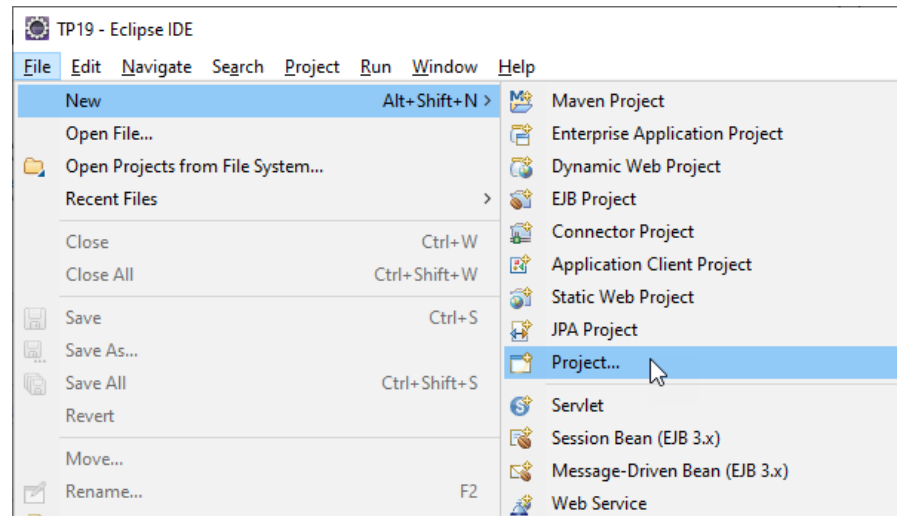
3.3. Servlet with Eclipse

1. Create a Dynamic web project (Method 2)



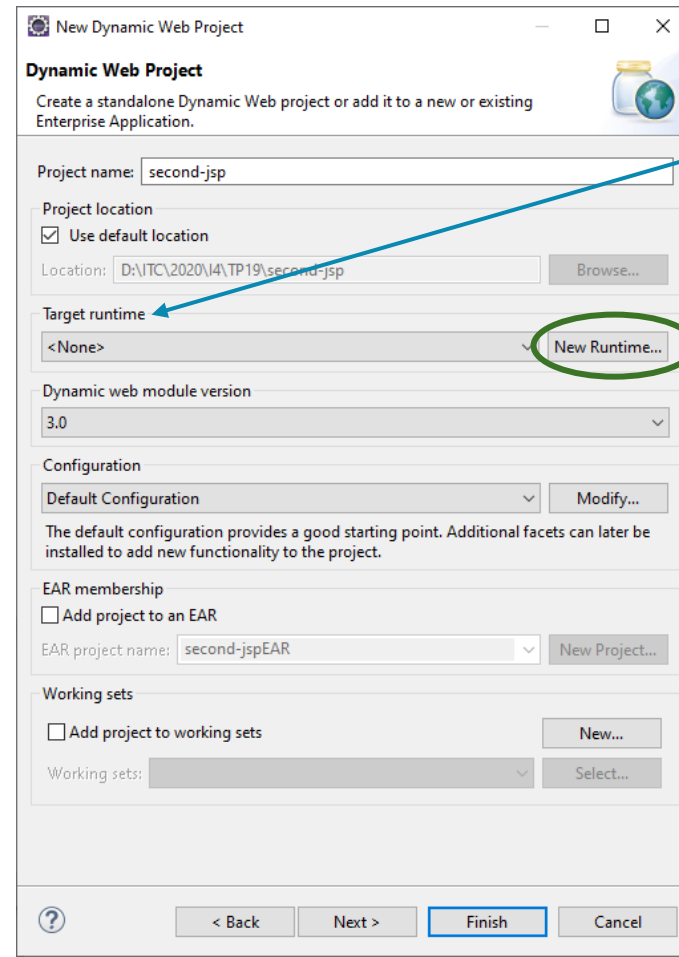
3.3. Servlet with Eclipse

1. Create a Dynamic web project (Method 3)

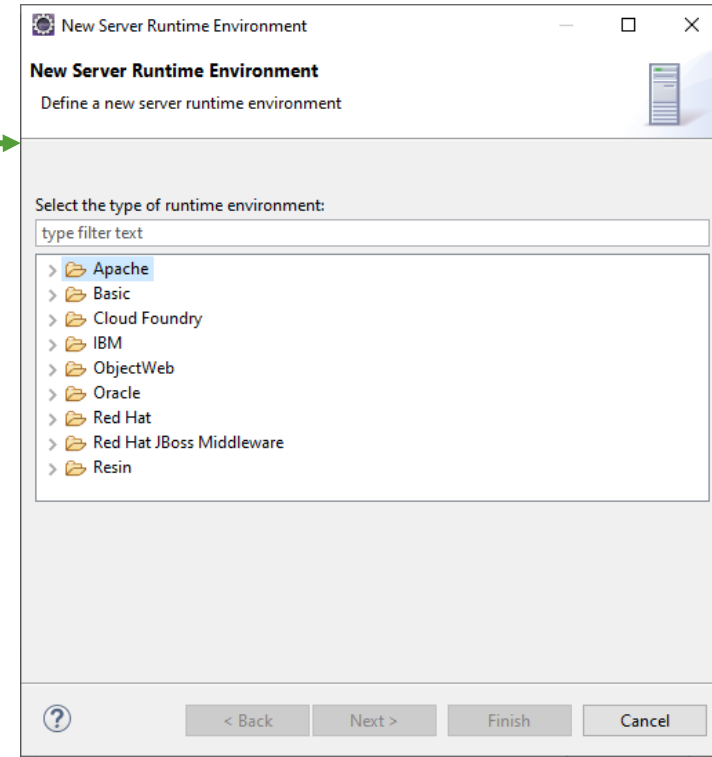


3.3. Servlet with Eclipse

1. Create a Dynamic web project (Step 2)

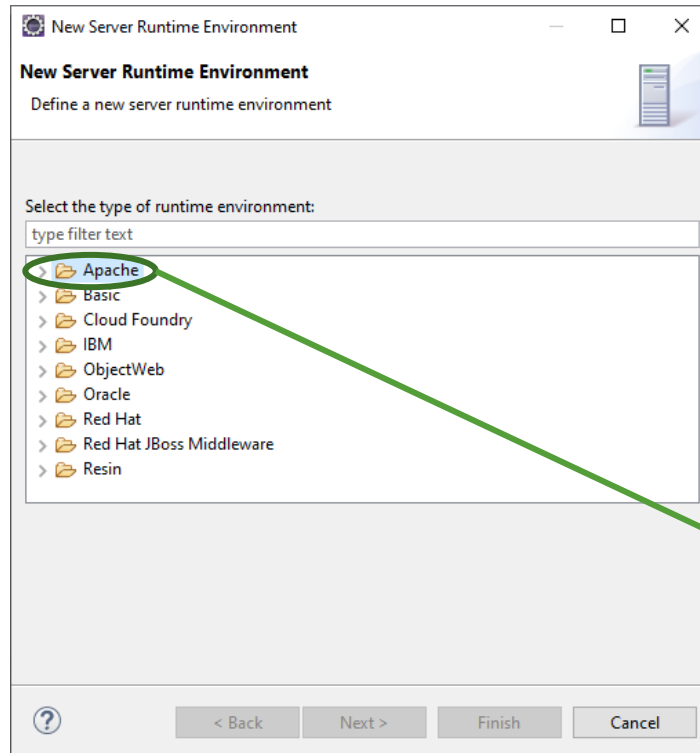


Target runtime: is server type you want to deploy to such as Apache Tomcat, Jetty, GlassFish, ...

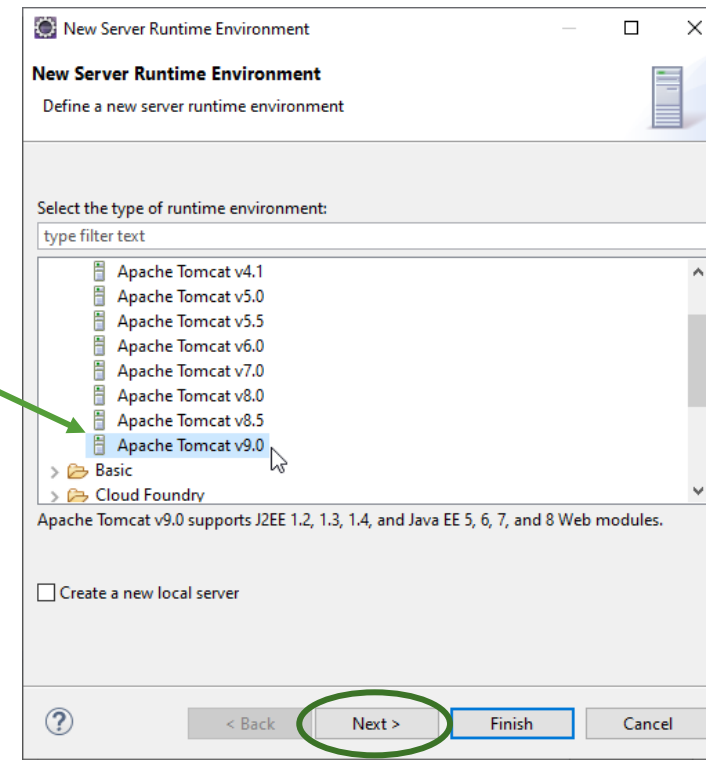


3.3. Servlet with Eclipse

1. Create a Dynamic web project (Step 2)

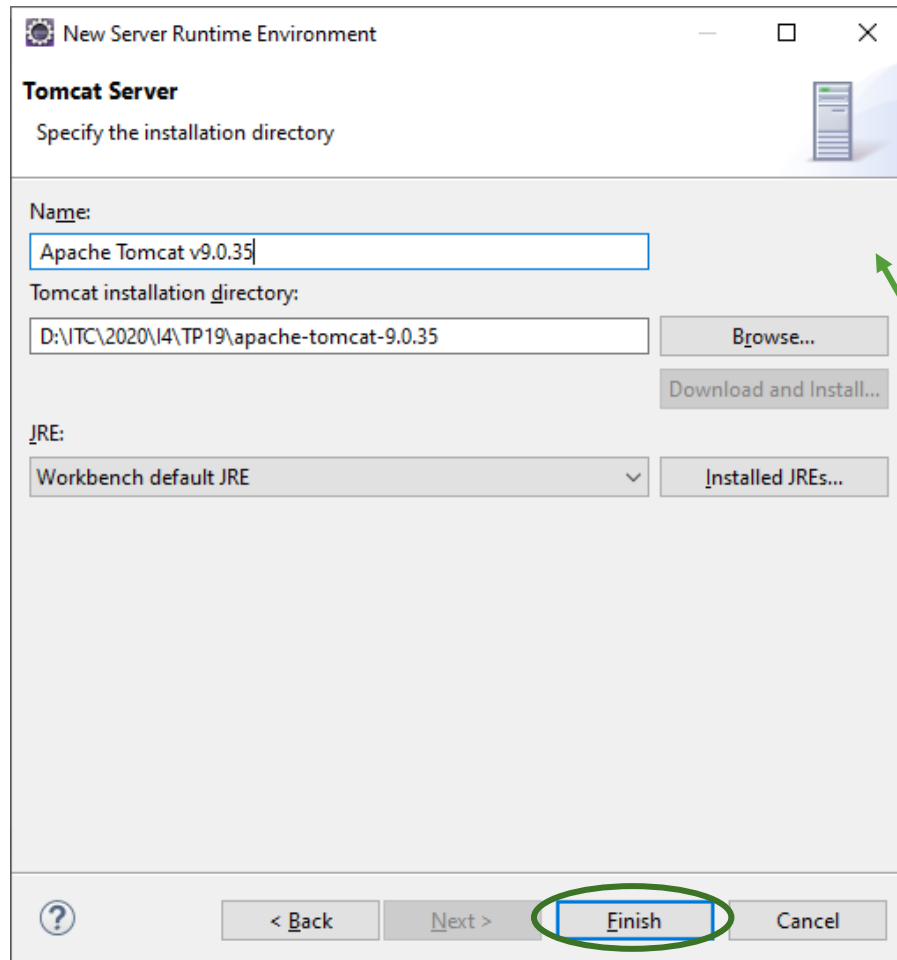


Select Target runtime as **Apache Tomcat v9.0** (because we downloaded last time)

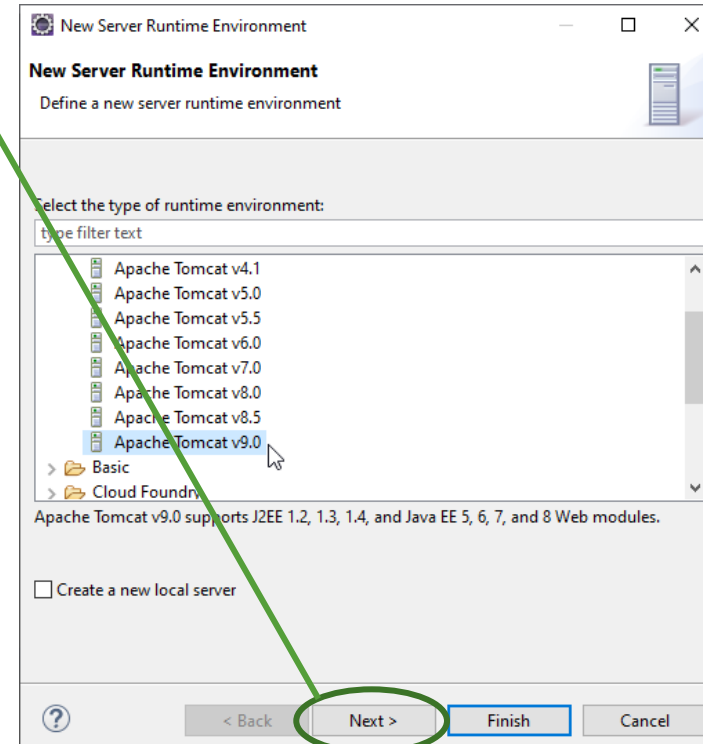


3.3. Servlet with Eclipse

1. Create a Dynamic web project (Step 2)



Select Target runtime as **Apache Tomcat v9.0** (because we downloaded last time)



3.3. Servlet with Eclipse

1. Create a Dynamic web project (Step 2)

New Dynamic Web Project

Dynamic Web Project
Create a standalone Dynamic Web project or add it to a new or existing Enterprise Application.

Project name:

Project location
☒ Use default location
Location:

Target runtime

Dynamic web module version

Configuration

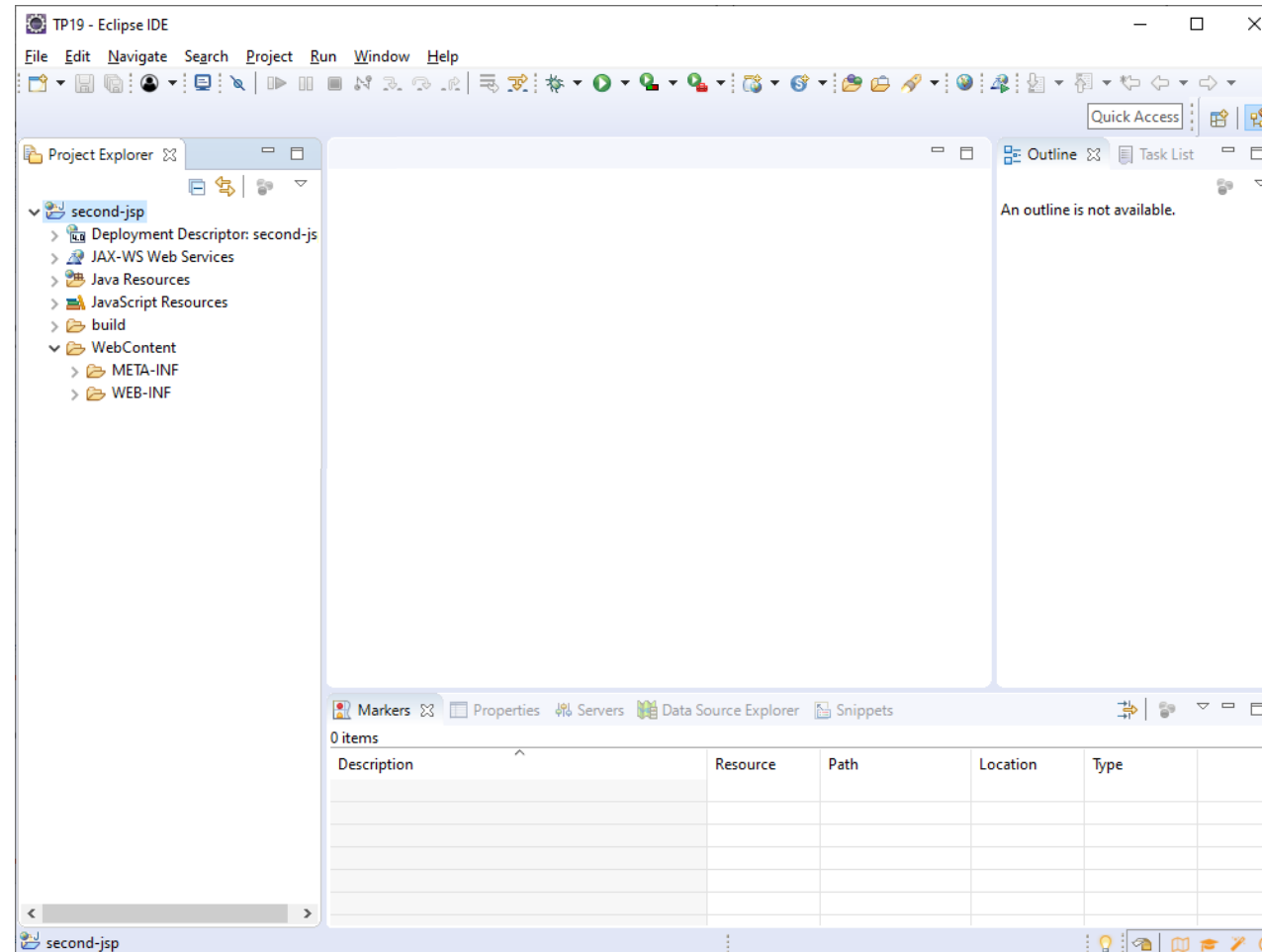
A good starting point for working with Apache Tomcat v9.0.35 runtime. Additional facets can later be installed to add new functionality to the project.

EAR membership
☐ Add project to an EAR
EAR project name:

Working sets
☐ Add project to working sets
Working sets:

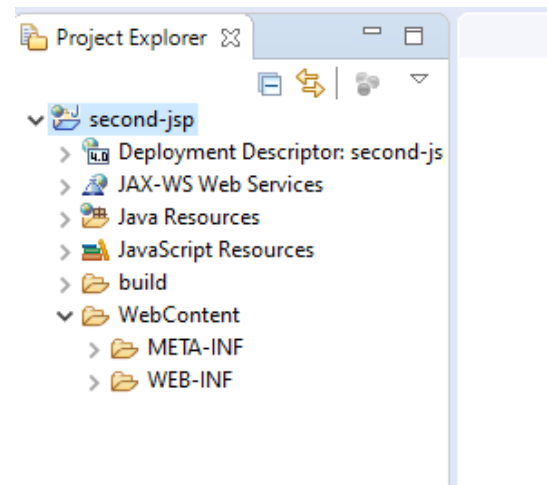
3.3. Servlet with Eclipse

1. Create a Dynamic web project



3.3. Servlet with Eclipse

1. Create a Dynamic web project



Deployment Descriptor: is web.xml visual editor.

JAX-WS Web Services: is for Web Services configurations

Java Resources: it is where you put Java source codes, packages and libraries

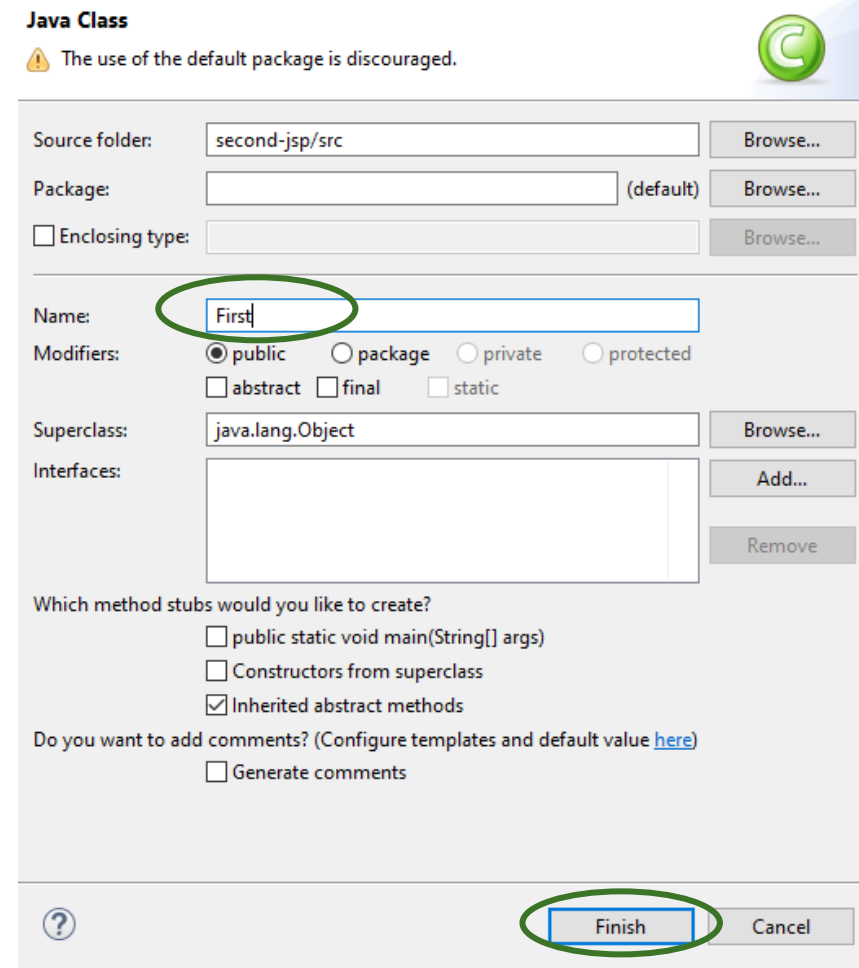
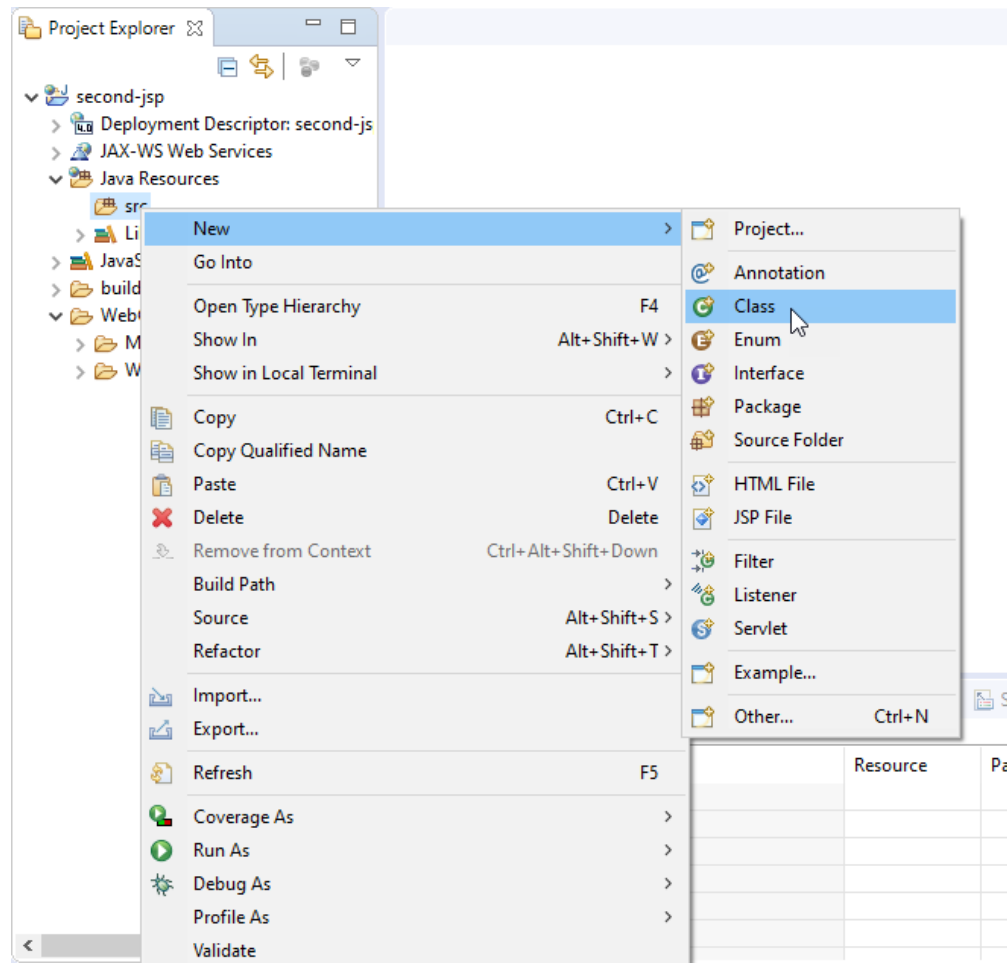
JavaScript Resources: it is where you put your JavaScript codes

build: is web module built (*.war,...)

WebContent: is web root folder and it is used to put HTML, CSS, JS, Configs, *.classes files.

3.3. Servlet with Eclipse

2. Create a Servlet - "First.java"





3.3. Servlet with Eclipse

2. Create a Servlet - "First.java"

```
import java.io.*;

import javax.servlet.*;

public class First implements Servlet {
    ServletConfig config = null;

    public void init(ServletConfig config) {
        this.config = config;
        System.out.println("servlet is initialized");
    }

    public void service(ServletRequest req, ServletResponse res)
        throws IOException, ServletException {

        res.setContentType("text/html");

        PrintWriter out = res.getWriter();
        out.print("<html><body>");
        out.print("<b>hello simple servlet</b>");
        out.print("</body></html>");
    }

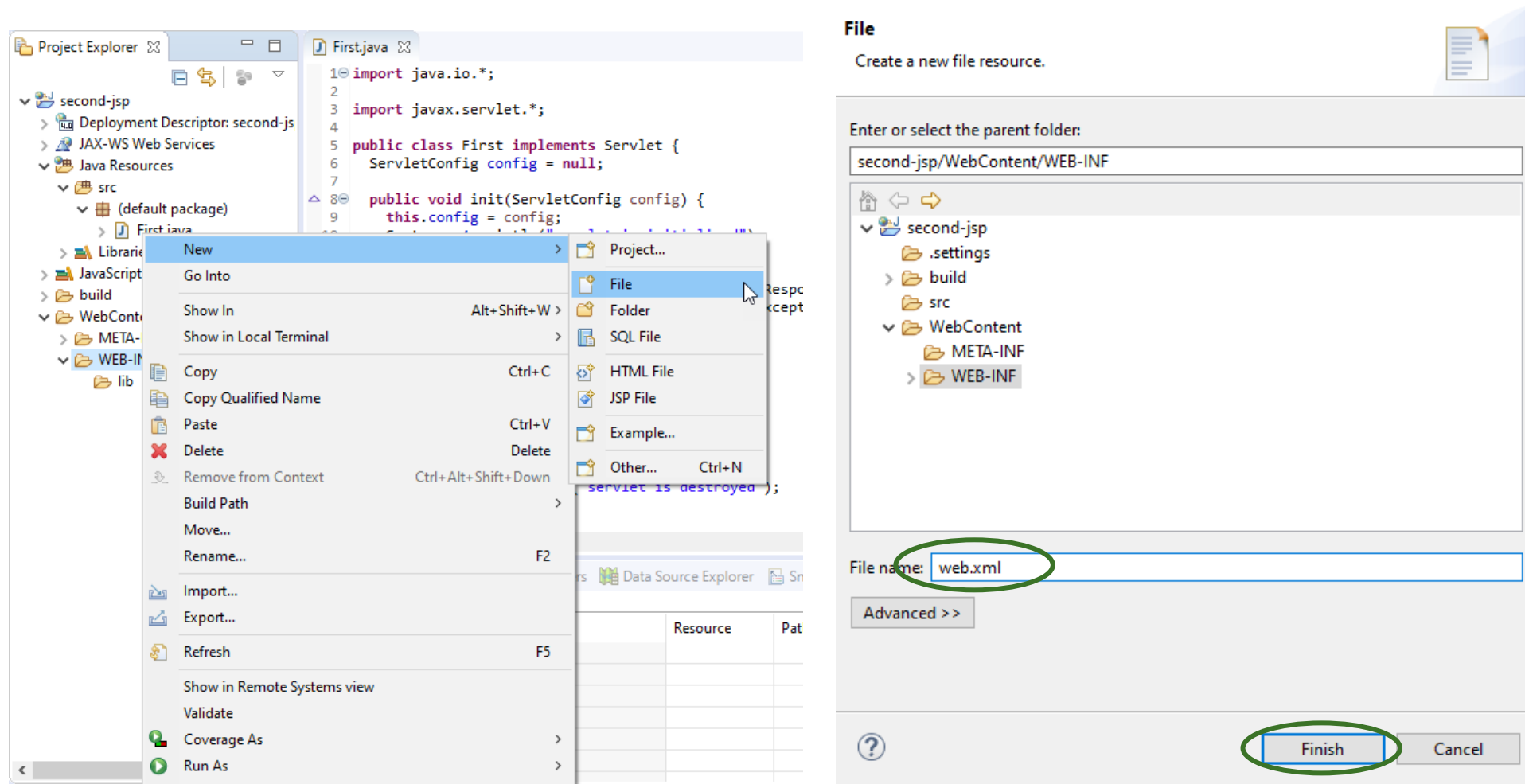
    public void destroy() {
        System.out.println("servlet is destroyed");
    }

    public ServletConfig getServletConfig() {
        return config;
    }

    public String getServletInfo() {
        return "copyright 2007-1010";
    }
}
```

3.3. Servlet with Eclipse

2. Create a Servlet - "web.xml"



The screenshot illustrates the process of creating a new file resource in Eclipse. The **Project Explorer** on the left shows the project structure, including the **WEB-INF** folder. The **File** menu is open, and the **New** option is selected, leading to the **Create a new file resource** dialog. The dialog shows the parent folder path **second-jsp/WebContent/WEB-INF** and the file name **web.xml**. The **Finish** button is highlighted.

Project Explorer

- second-jsp
 - Deployment Descriptor: second-js
 - JAX-WS Web Services
 - Java Resources
 - src
 - (default package)
 - First.java
 - Libraries
 - JavaScript
 - build
 - WebContent
 - META-INF
 - WEB-INF

File

Create a new file resource.

Enter or select the parent folder:

second-jsp/WebContent/WEB-INF

second-jsp

- .settings
- build
- src
- WebContent
 - META-INF
 - WEB-INF

File name: web.xml

Advanced >>

Finish Cancel

3.3. Servlet with Eclipse

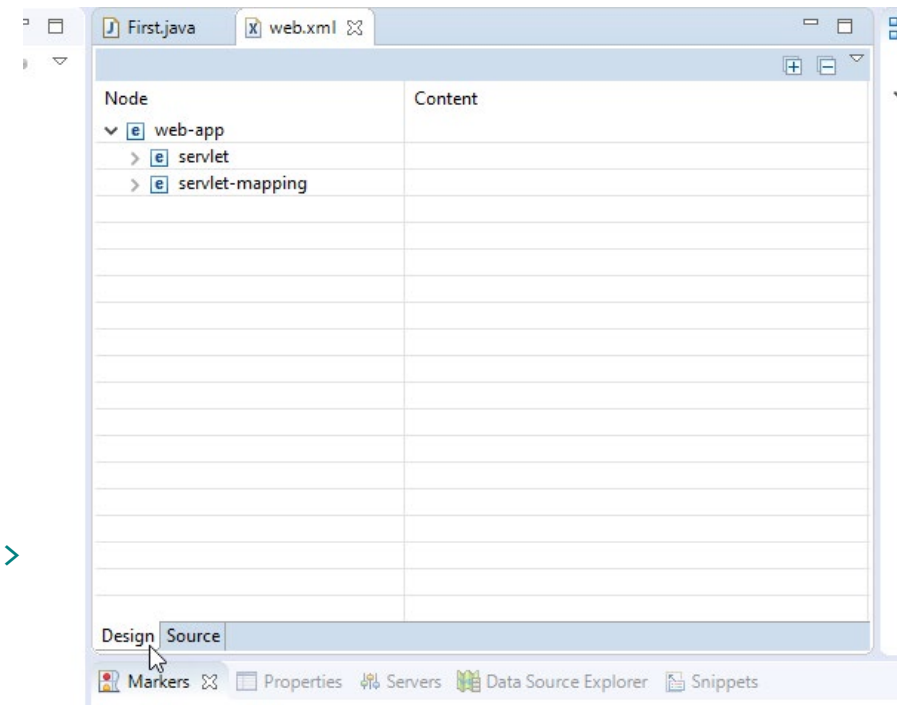
2. Create a Servlet - "web.xml"

```
<web-app>

    <servlet>
        <servlet-name>my-first-servlet</servlet-name>
        <servlet-class>First</servlet-class>
    </servlet>

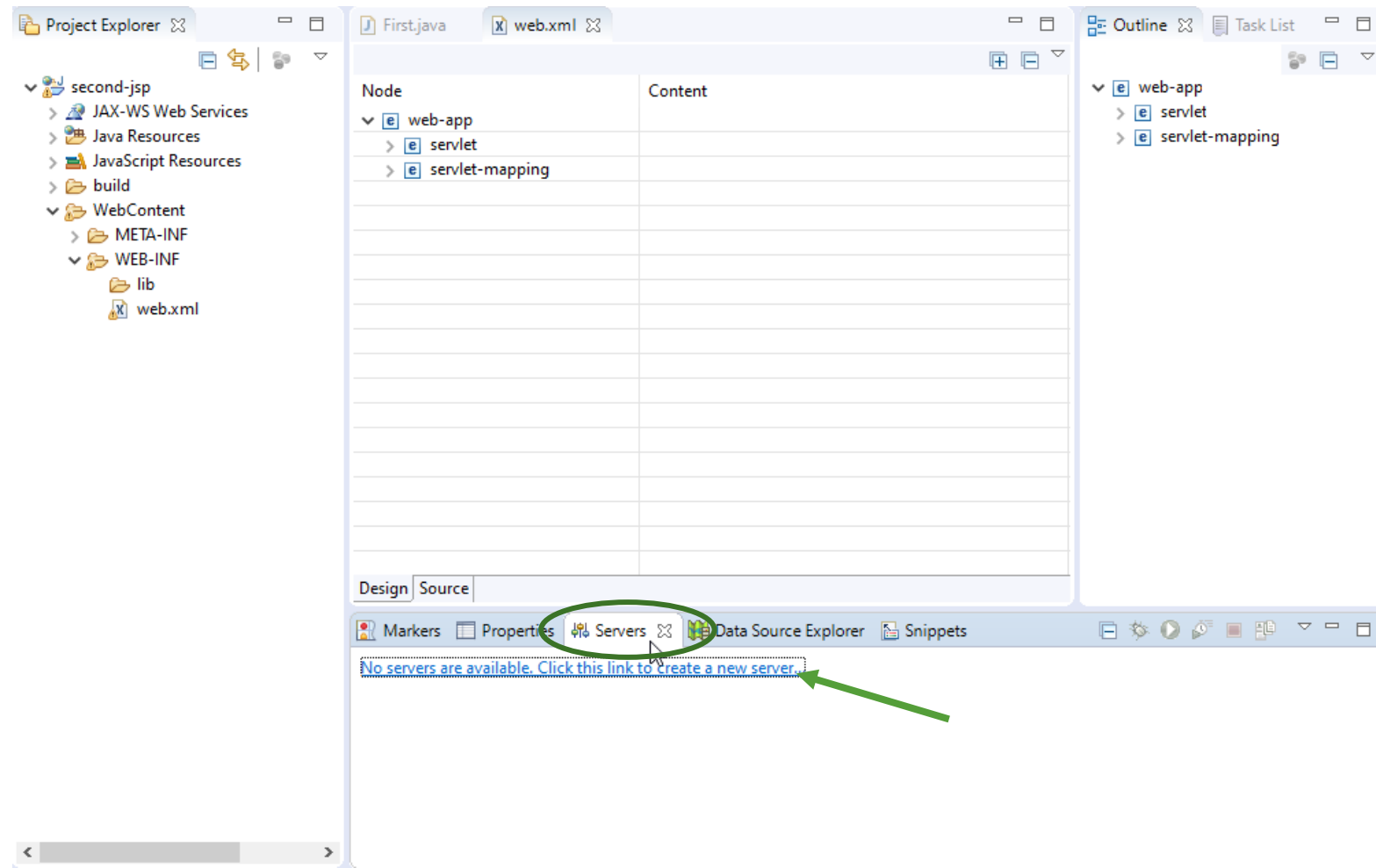
    <servlet-mapping>
        <servlet-name>my-first-servlet</servlet-name>
        <url-pattern>/welcome</url-pattern>
    </servlet-mapping>

</web-app>
```



3.3. Servlet with Eclipse

3. Run Servlet



3.3. Servlet with Eclipse

3. Run Servlet – Add Server

Define a New Server

Choose the type of server to create

Select the server type:

type filter text

- Tomcat v5.5 Server
- Tomcat v6.0 Server
- Tomcat v7.0 Server
- Tomcat v8.0 Server
- Tomcat v8.5 Server
- Tomcat v9.0 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 v9.0.35 Server at localhost

Server runtime environment:

Apache Tomcat v9.0

Add...

[Configure runtime environments...](#)



< Back

Next >

Finish

Cancel

Add and Remove

Modify the resources that are configured on the server

Move resources to the right to configure them on the server

Available:

second-jsp

Configured:

Add >

< Remove

Add All >>

<< Remove All



< Back

Next >

Finish

Cancel

3.3. Servlet with Eclipse

3. Run Servlet – Add Server

Define a New Server

Choose the type of server to create

Select the server type:

type filter text

- Tomcat v5.5 Server
- Tomcat v6.0 Server
- Tomcat v7.0 Server
- Tomcat v8.0 Server
- Tomcat v8.5 Server
- Tomcat v9.0 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 v9.0.35 Server at localhost

Server runtime environment:

Apache Tomcat v9.0

Add...

[Configure runtime environments...](#)



< Back

Next >

Finish

Cancel

Add and Remove

Modify the resources that are configured on the server

Move resources to the right to configure them on the server

Available:

Configured:

second-jsp

Add >

< Remove

Add All >>

<< Remove All



< Back

Next >

Finish

Cancel

3.3. Servlet with Eclipse

3. Run Servlet – Start Server

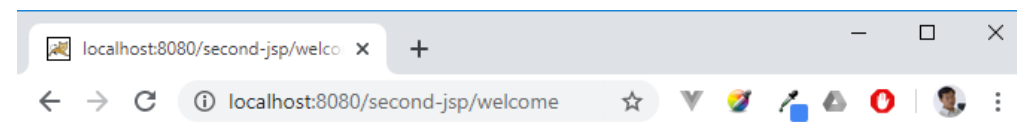
The screenshot shows the Eclipse IDE interface. On the left, the Project Explorer shows a project named 'second-jsp' with sub-items: JAX-WS Web Services, Java Resources, JavaScript Resources, build, WebContent, and Servers. The main editor shows 'First.java' and 'web.xml'. The 'Servers' tab in the bottom toolbar is circled in green, showing 'tomcat v9.0.35 Server at localhost [Stopped, Republish]' and 'second-jsp'. A Windows Security Alert dialog box is open, titled 'Windows Security Alert'. It states: 'Windows Defender Firewall has blocked some features of this app'. The app is 'Java(TM) Platform SE binary' by Oracle Corporation, located at 'C:\program files\java\jdk1.8.0_181\bin\javaw.exe'. It asks to allow communication on networks: Private networks (unchecked) and Public networks (checked). The 'Allow access' button is circled in green. A green arrow points from the 'Run' button in the Eclipse toolbar to the 'Allow access' button. The 'Run' button is also circled in green. A tooltip 'Start the server (Ctrl+Alt+R)' is visible near the 'Run' button.

3.3. Servlet with Eclipse

3. Run Servlet – Check Server Status

```
Tomcat v9.0.35 Server at localhost [Apache Tomcat] C:\Program Files\Java\jdk1.8.0_181\bin\java.exe (May 21, 2020, 1:06:02 AM)
May 21, 2020 1:06:05 AM org.apache.catalina.startup.Catalina load
INFO: Server initialization in [2,095] milliseconds
May 21, 2020 1:06:05 AM org.apache.catalina.core.StandardService startInternal
INFO: Starting service [Catalina]
May 21, 2020 1:06:05 AM org.apache.catalina.core.StandardEngine startInternal
INFO: Starting Servlet engine: [Apache Tomcat/9.0.35]
May 21, 2020 1:06:10 AM org.apache.jasper.servlet.TldScanner scanJars
INFO: At least one JAR was scanned for TLDs yet contained no TLDs. Enable debug logging for this logger for a complete list of JARs that were scanned but no TLDs were
May 21, 2020 1:06:14 AM org.apache.jasper.servlet.TldScanner scanJars
INFO: At least one JAR was scanned for TLDs yet contained no TLDs. Enable debug logging for this logger for a complete list of JARs that were scanned but no TLDs were
May 21, 2020 1:06:14 AM org.apache.coyote.AbstractProtocol start
INFO: Starting ProtocolHandler ["http-nio-8080"]
May 21, 2020 1:06:14 AM org.apache.catalina.startup.Catalina start
INFO: Server startup in [9,017] milliseconds
```

Server is correctly
started up at port
8080.



Now open browser and navigate to
<http://localhost:8080/second-jsp/welcome>



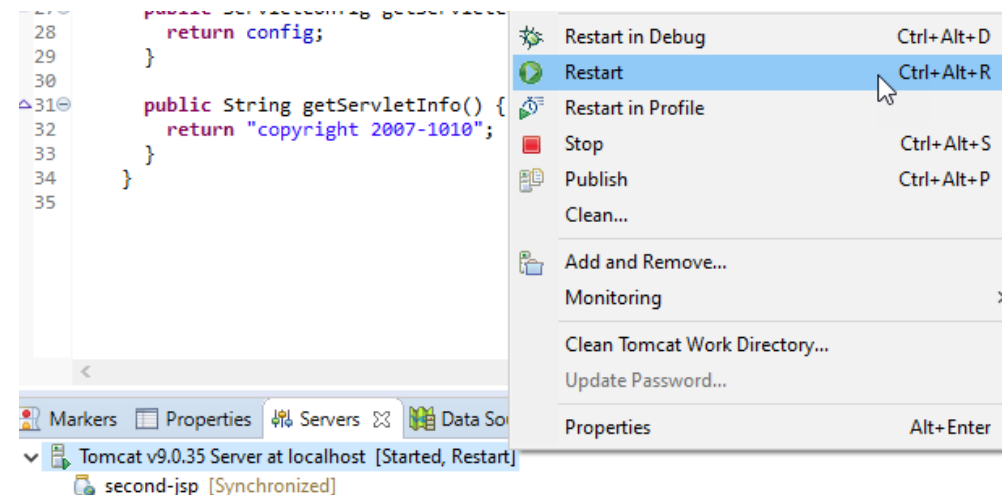
3.3. Servlet with Eclipse

3. Run Servlet – Updating codes

Now try to change some codes in First.java and then save it. You will notice the console says:

```
May 21, 2020 1:18:44 AM org.apache.catalina.core.StandardContext reload
INFO: Reloading Context with name [/second-jsp] has started
servlet is destroyed
May 21, 2020 1:18:49 AM org.apache.jasper.servlet.TldScanner scanJars
INFO: At least one JAR was scanned for TLDs yet contained no TLDs. Enable debug logg:
May 21, 2020 1:18:49 AM org.apache.catalina.core.StandardContext reload
INFO: Reloading Context with name [/second-jsp] is completed
```

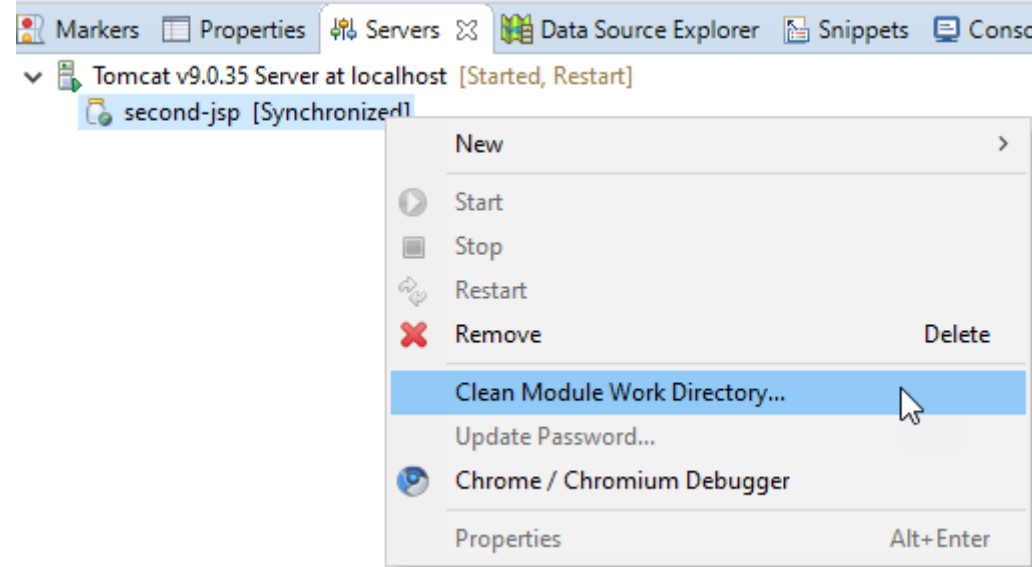
But sometimes it v reload, so we need restart the server.



3.3. Servlet with Eclipse

3. Run Servlet – Updating codes

Sometimes when we rename class or restructure classes, we need clean cache:





References

- <https://www.javatpoint.com/steps-to-create-a-servlet-using-tomcat-server>
- <https://www.javatpoint.com/jsp-tutorial>
- <https://www.javatpoint.com/servlet-tutorial>
- <https://www.javatpoint.com/Servlet-interface>