

CH01_第一个Java Web应用



一、什么是Web?

现今，Web是网络上使用最广泛的分布式应用架构。那什么是Web呢？Web是一种分布式应用架构，是为了共享分布在网络上的各个Web服务器中所有互相连接的信息。Web采用“客户端/服务器”的通信模式，客户端与服务器之间用HTTP协议通信。Web使用超级文本（HTML）来连接网络上的信息。信息存放在服务器端，客户端通过浏览器（例如IE或Chrome）就可以查找网络中各个Web服务器上的信息。

目前，与Web相关的各种技术标准都由著名的W3C组织（WorldWideWebConsortium）来管理和维护。

人们还习惯把www称为Web，www全称World Wide Web，中文名万维网。简单说www是指全球范围内的Web。www以Internet作为网络平台，Internet是来自世界各地的众多相互连接的计算机和其他设备的集合，也就是我们说的互联网。

万维网并不等同互联网，万维网只是互联网所能提供的服务其中之一，是靠着互联网运行的一项服务。

二、Web中的一些概念

1. URL

当我们上网的时候，输入网址，比如<https://www.w3school.com.cn/html/index.asp>，我们就能看到W3school学习网站，网站上的信息是Web服务器发送过来的。这个网址就是URL，全称Uniform Resource Locator，中文名统一资源定位符，是为了标记网络上资源的位置而设的一种编址方式。

URL一般由三部分构成，即协议类型、主机IP或者域名、资源路径。上面网址中，https是协议类型，www.w3school.com.cn是域名，html/index.asp是资源路径。

2. HTML

HTML全称HyperTextMarkupLanguage，中文名超文本标记语言。主要用来把Web服务器的信息显示给用户，这些信息包括文字、图片、动画、音频和视频等等。

HTML从上世纪90年代发展到现在的HTML5，功能越来越强大。

3. HTTP和HTTPS

- HTTP全称Hypertext Transfer Protocol，中文名超文本传输协议。是一种规定如何在网络中传输超级文本的协议，它规定了Web的基本运作过程，以及浏览器与Web服务器之间的通信细节。

HTTP采用客户端/服务器的通信模式，服务器端一般问Web服务器，客户端一般为浏览器。

- HTTPS全称Hypertext Transfer Protocol Secure，中文名超文本传输安全协议。是以安全为目标的HTTP协议，简单讲是HTTP协议的安全版。HTTP协议是以明文来传输，而HTTPS采用SSL加密来传输。

三、创建第一个Java Web应用

1. 什么是Java Web应用？

Oracle公司的Servlet规范对JavaWeb应用做了这样的定义：“JavaWeb应用由一组Servlet/JSP、HTML文件、相关Java类，以及其他可以被绑定的资源构成。它可以在由各种供应商提供的符合Servlet规范的Servlet容器中运行。”从JavaWeb应用的定义可以看出，JavaWeb应用不仅可以在Tomcat中运行，还可以在其他符合Servlet规范的Servlet容器中运行。

从这段定义可以看出，Java Web应用不仅仅可以在Tomcat中运行，还可以在任何符合Servlet规

范的Servlet容器中运行。

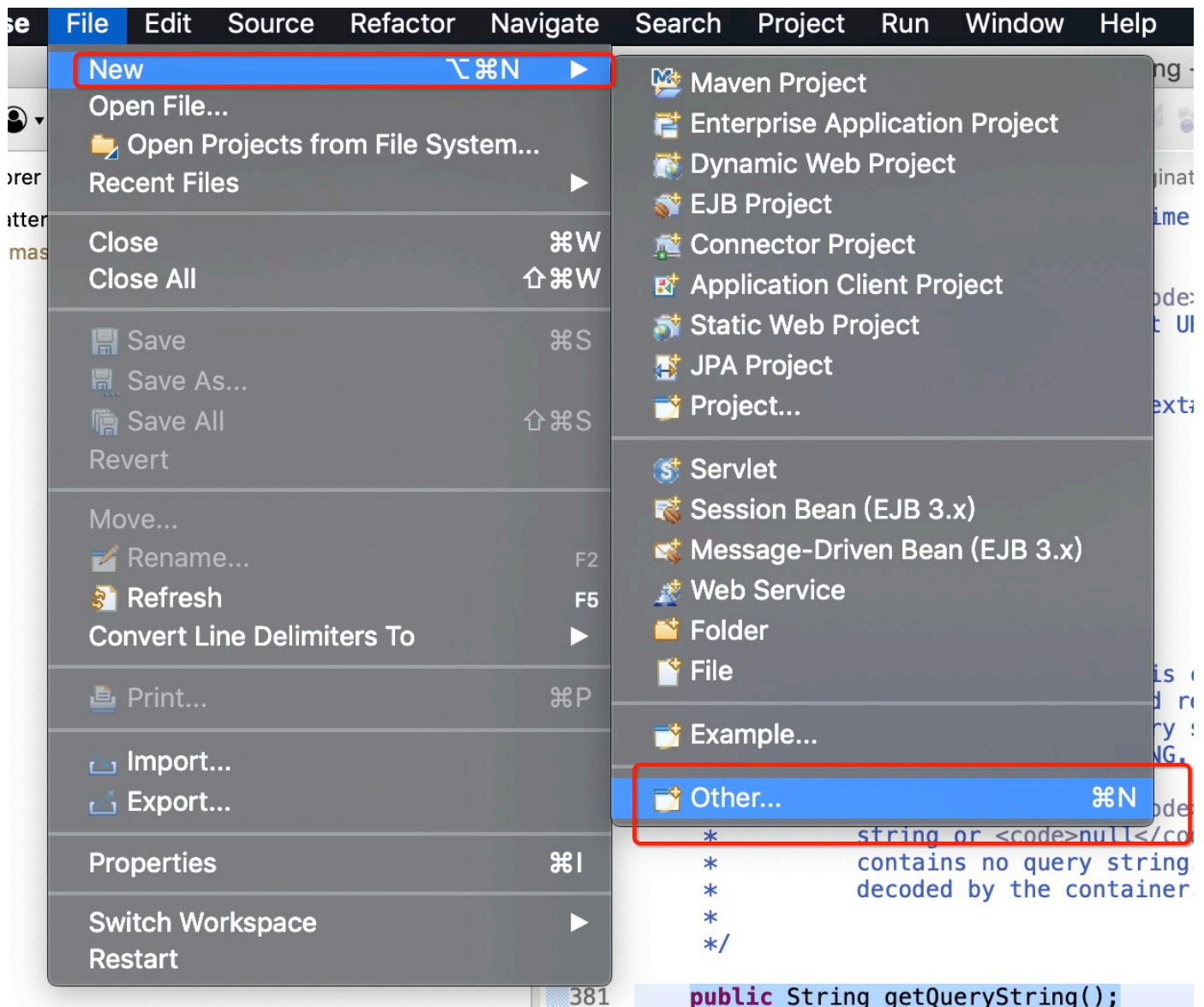
2. Java Web应用包含的内容

- Servlet组件: Servlet接口的实现类，运行在服务器，包含了被Servlet容器动态调用的代码。
- JSP: 全称Java Server Page，包含Java程序代码的html文档，运行在服务器，当客户端请求访问JSP时，Servlet容器先把它编译成Servlet类，然后动态调用它的程序代码。
- Java类: 开发者根据业务需求定义的和Web应用相关的Java类。
- 静态文档: 存放在服务器的文件系统中，如HTML文件、图片文件和声音文件等等。当客户端请求访问这些文件时，Servlet容器先从本地文件系统中读取这些文件的数据，再把它发送到客户端。
- 客户端脚本程序: 由客户端来运行的程序，比如JavaScript。
- web.xml文件: Java Web应用的配置文件，可以说是Web应用的大脑，该文件必须位于Java Web 应用的WEB-INF文件夹下。

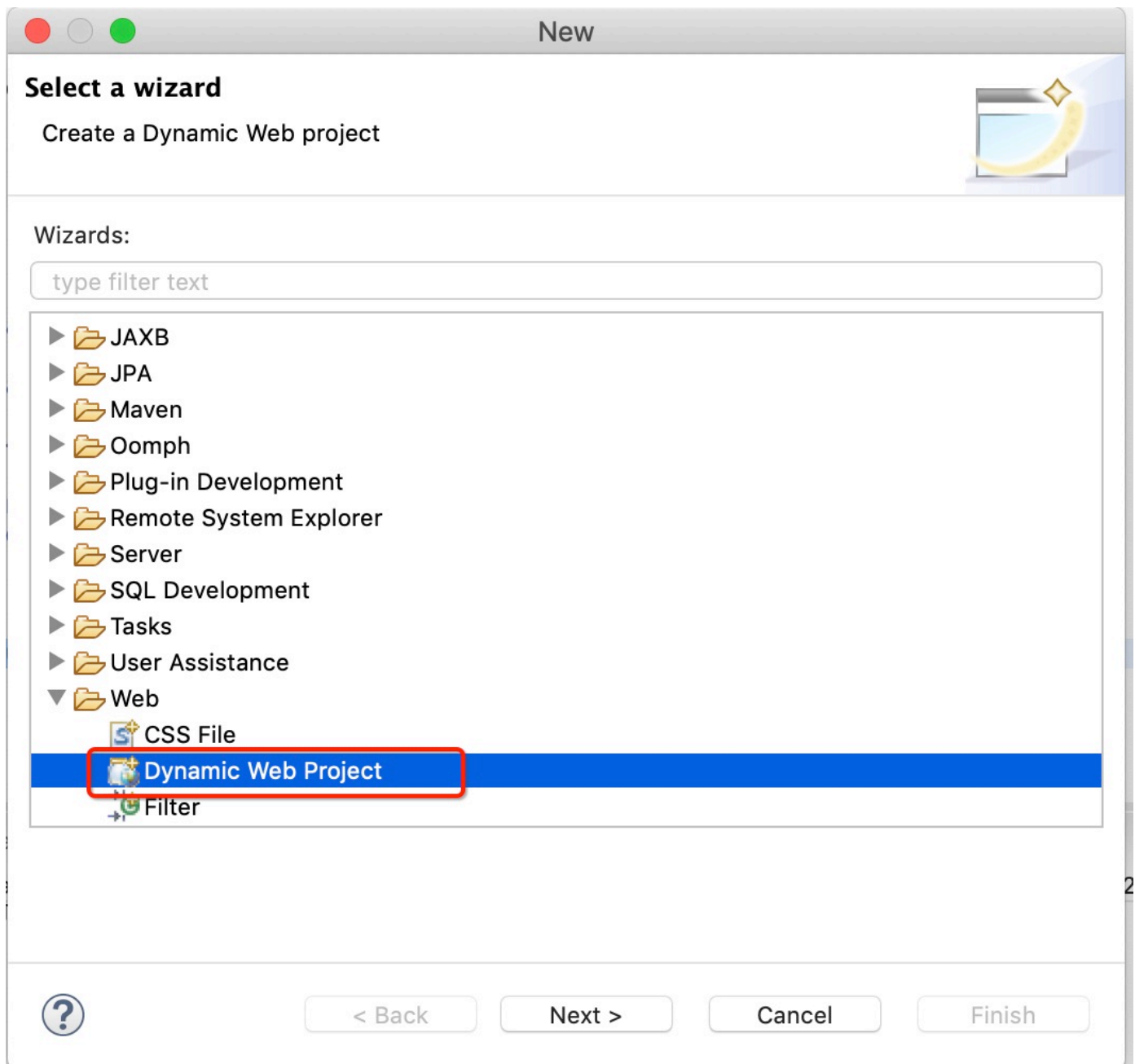
3. 如何创建Java Web项目？

为了方便我们直接在eclipse下创建web项目，项目名称为first-web。参考下面:

- 点击eclipse菜单File => New => Other:



- 选择Dynamic Web Project, 点击Next:




- 输入项目名称first-web， 点击Finish:

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

Hint: Get started quickly by selecting one of the pre-defined project configurations.

EAR membership

☐ Add project to an EAR

EAR project name:

Working sets

☐ Add project to working sets

Working sets:

- 我们会看到有个WebContent的文件夹，我们在下面创建文件夹jsp、html、js、css分别用来存放jsp文件、html文件、JavaScript文件和css文件。目录结构如下：

- ▼ first-web
 - ▶ Deployment Descriptor: first-web
 - ▶ JAX-WS Web Services
 - ▼ Java Resources
 - ▼ src
 - ▼ com.hx.servlet
 - ▼ LoginServlet.java
 - ▶ LoginServlet
 - ▶ Libraries
 - ▶ JavaScript Resources
 - ▶ build

▼ WebContent

- ▼ css
 - bootstrap-theme.css
 - bootstrap-theme.css.map
 - bootstrap.css
 - bootstrap.css.map
- ▼ html
 - login.html
- ▼ js
 - bootstrap.js
 - jquery-1.12.js
- ▼ jsp
 - loginFailed.jsp
 - loginSuccess.jsp
- ▶ META-INF
- ▼ WEB-INF
 - lib
 - web.xml

- 第一个Web应用结构创建完毕，我们只需要在对应的目录下面创建代码实现功能。

- 在html文件夹下创建一个登陆的html文件，取名login.html。这里因为使用了Bootstrap，所以页面导入了Bootstrap的css和js文件。
- 在jsp文件夹下创建两个jsp文件，分别取名loginSuccess.jsp(该页面显示登陆成功过的信息)和loginFailed.jsp(该页面显示登陆失败的信息)。
- 我们在com.hx.servlet包下面定义一个LoginServlet类，该类继承了GenericServlet类，通过重写service方法来判断登陆的用户名、密码是否和我们预先设定的用户名、密码一致。如果一致则跳转去登陆成功页面loginSuccess.jsp；否则跳转去登陆失败页面loginFailed.jsp。
- 修改web.xml文件，在其中配置我们创建的servlet。

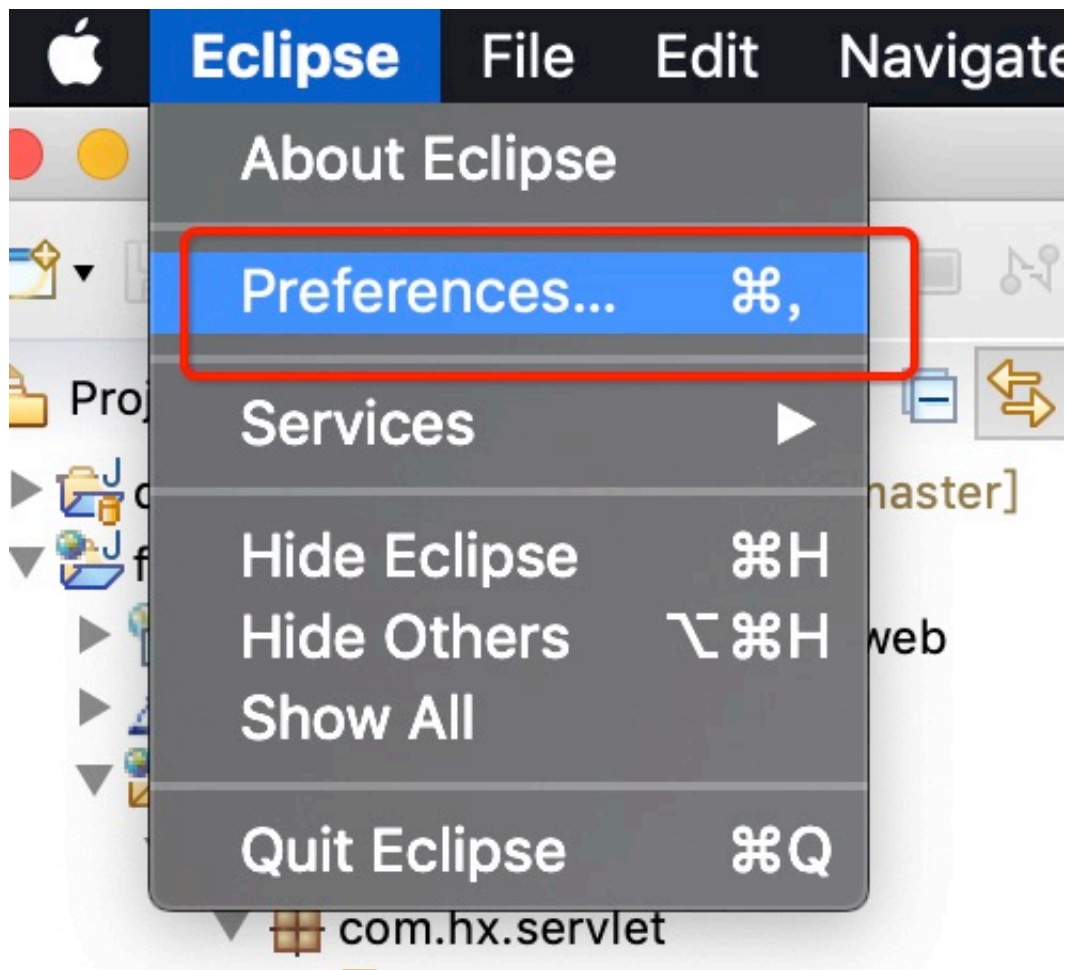
以上就是一个简单的Web项目，完整代码参考 [第一个Web项目代码](#)。

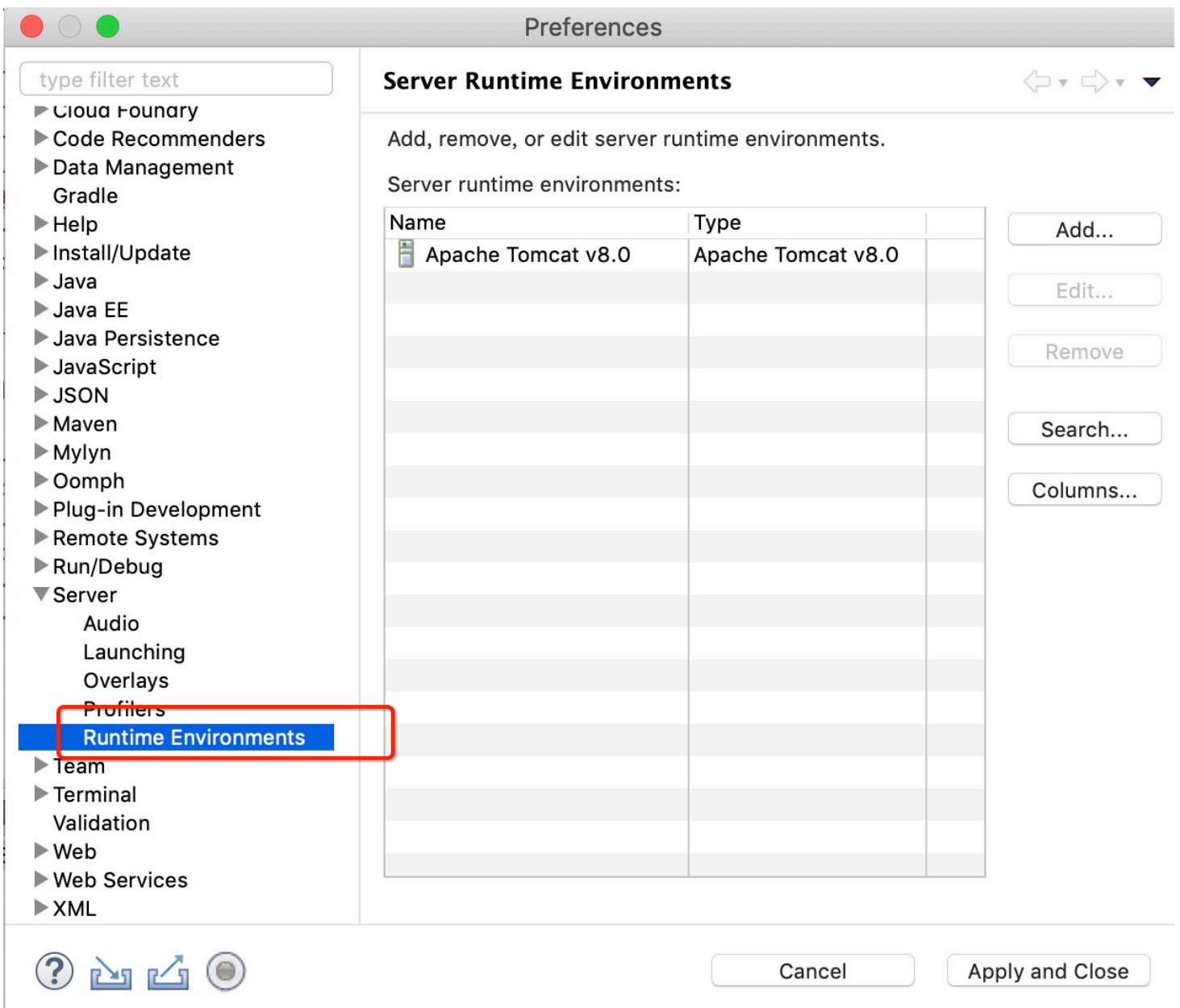
注意: 客户端不能直接请求访问WEB-INF目录下的文件，该目录下的文件只能被服务端访问。

4. 部署运行第一个Web项目

Web项目创建完了，我们需要一个Web服务器来运行它。这里我们用Tomcat，Tomcat需要安装和配置，参考 [Tomcat安装和配置指南](#)。

为了后续学习方便，我们把刚刚安装的Tomcat集成到eclipse中，参考下面步骤:





点击Add，选择对应的Tomcat版本

New Server Runtime Environment


New Server Runtime Environment


Define a new server runtime environment





Select the type of runtime environment:


type filter text


▼  Apache


 Apache Tomcat v3.2


 Apache Tomcat v4.0


 Apache Tomcat v4.1


 Apache Tomcat v5.0


 Apache Tomcat v5.5

 Apache Tomcat v6.0

 Apache Tomcat v7.0

 Apache Tomcat v8.0

 Apache Tomcat v8.5

 Apache Tomcat v9.0

Apache Tomcat v8.0 supports J2EE 1.2, 1.3, 1.4, and Java EE 5, 6, and 7 Web modules.

☐ Create a new local server



< Back

Next >

Cancel

Finish

选择Tomcat的安装路径，点击Finish

New Server Runtime Environment

Tomcat Server

Specify the installation directory

Name:
Apache Tomcat v8.0 (2)

Tomcat installation directory:
/Users/j2ee/server/apache-tomcat-8.0.53

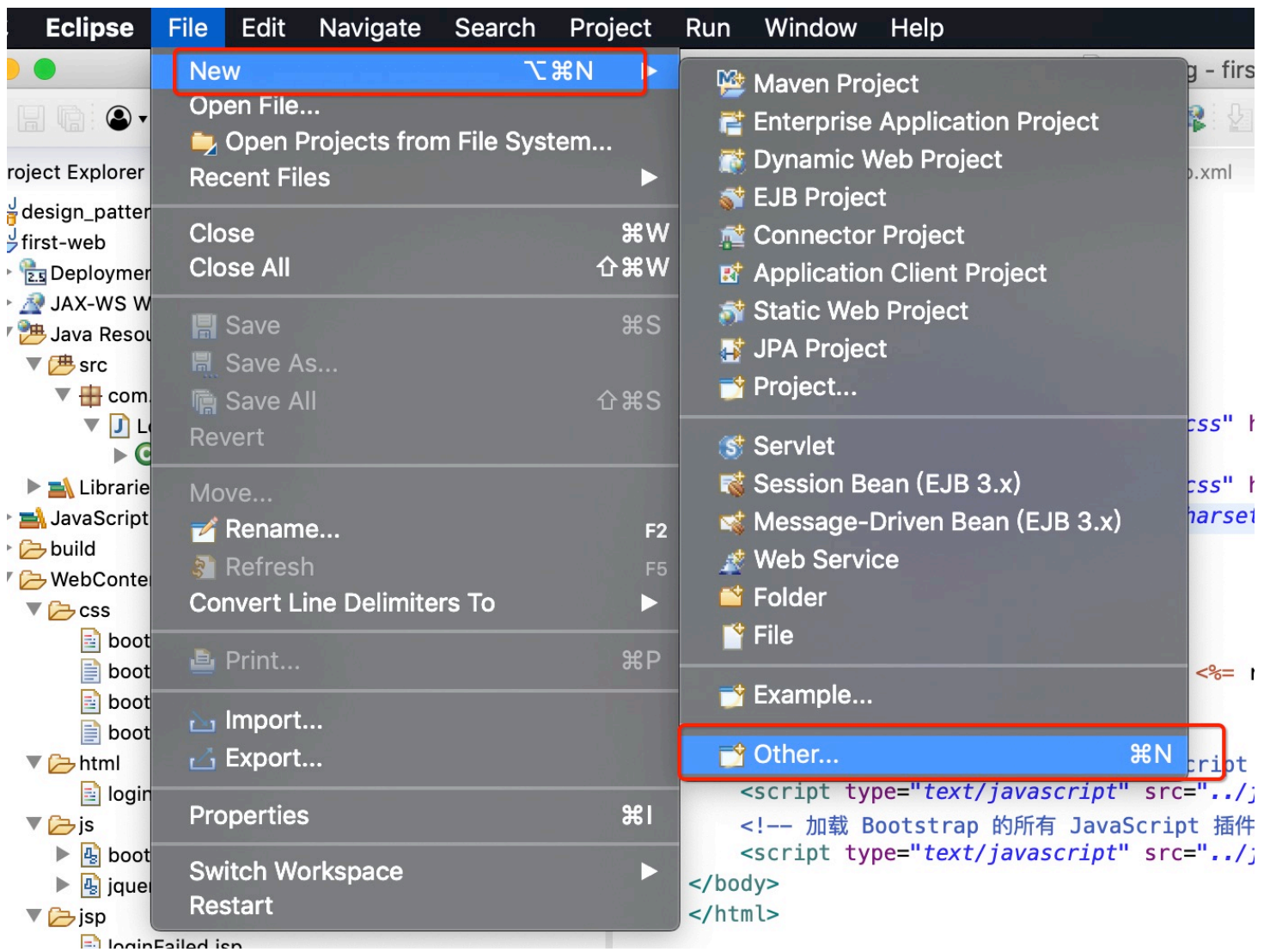
Browse...
Download and Install...
Installed JREs...

JRE:
Workbench default JRE

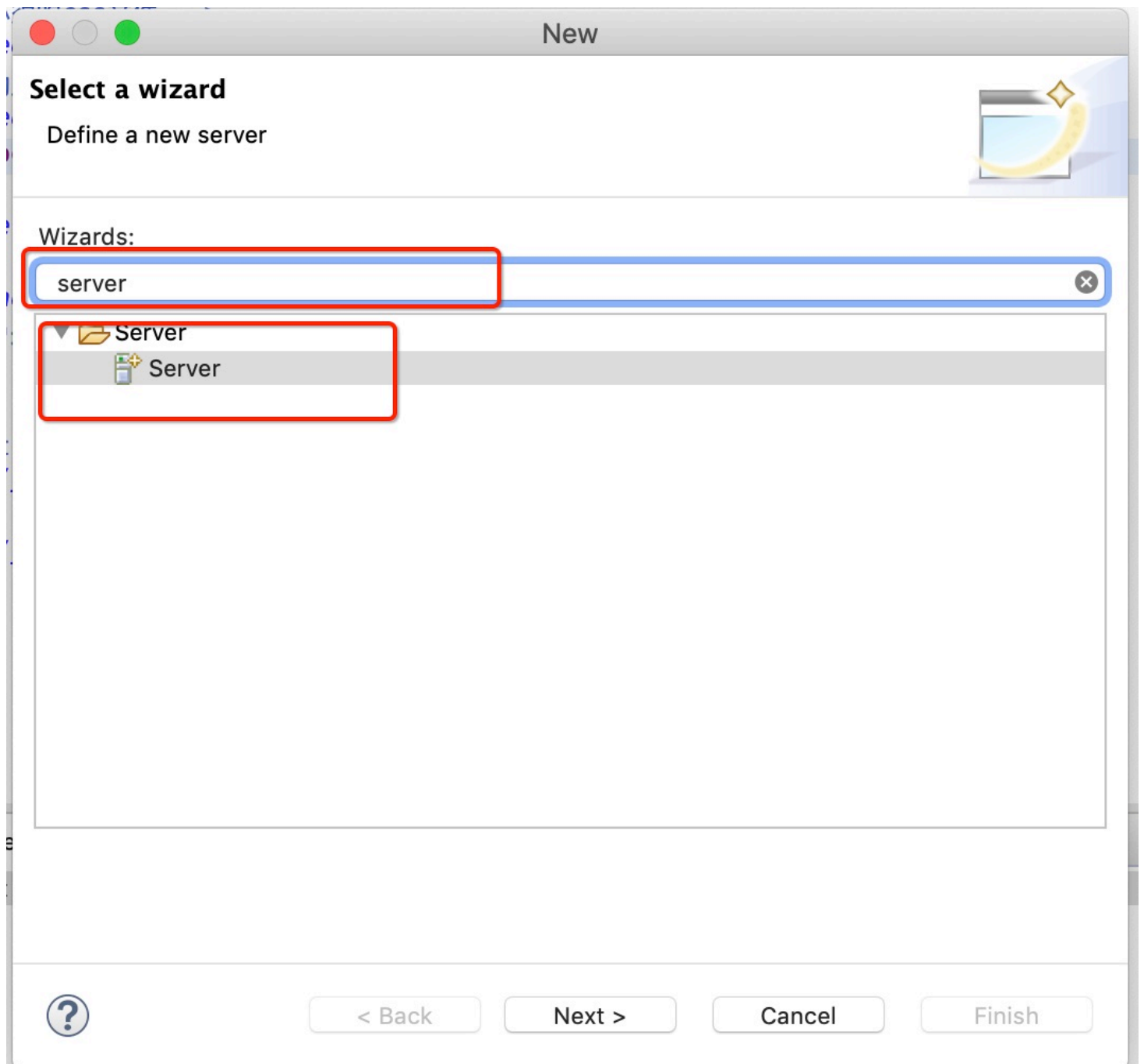
选择你的Tomcat
安装路径

? < Back Next > Cancel Finish

继承Tomcat完毕，我们在eclipse中创建一个server:



选中server, 点击Next:




点击Finish，server创建完毕：

New Server


Define a New Server


Choose the type of server to create





Select the server type:


type filter text


 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


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 v8.0




Server runtime environment:

Apache Tomcat v8.0



[Add...](#)

[Configure runtime environments...](#)



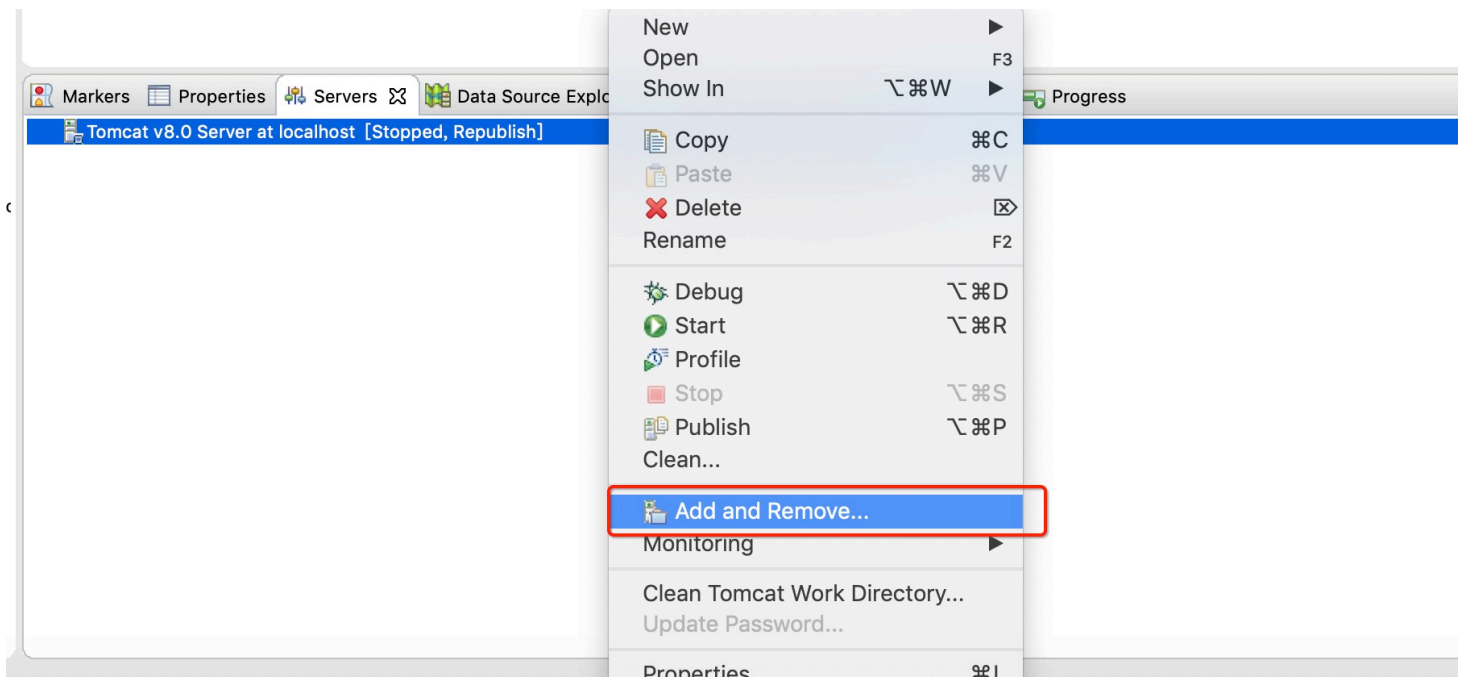
< Back

Next >

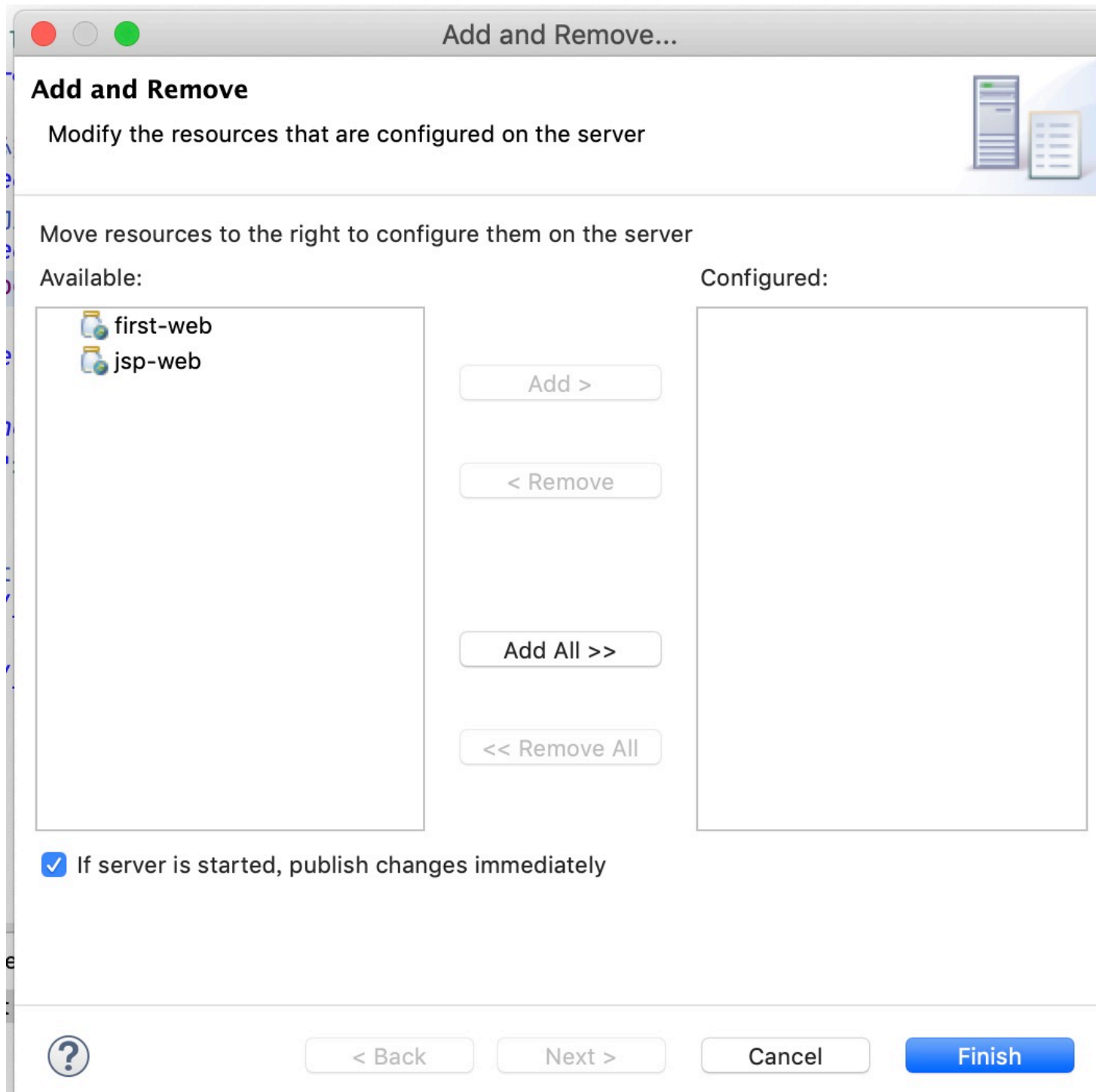
Cancel

Finish

我们会在server视图下面有我们刚刚创建的server，我们把第一个Web项目部署到Tomcat中：

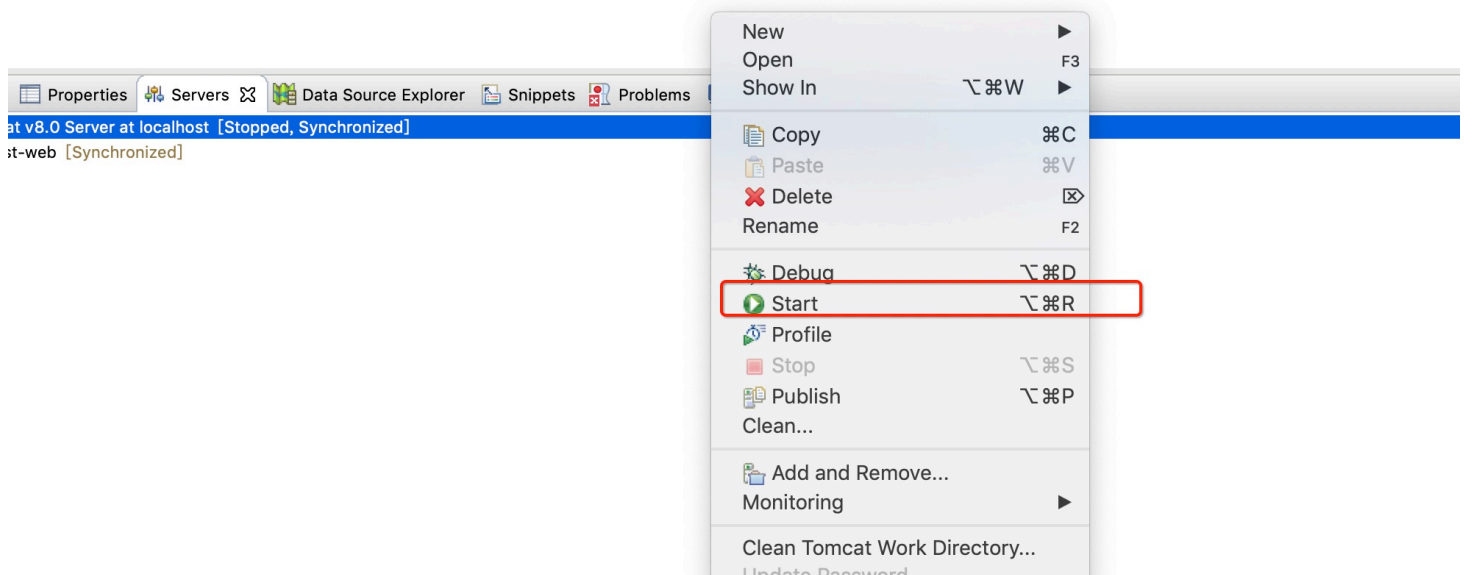


选中需要部署的项目，点击Add，最后点击Finish:



下面启动Tomcat:

在创建的server上右键点击，点击Start，Tomcat将会启动。



在浏览器中输入<http://localhost:8080/first-web/>，访问项目，我们会发现显示的是login.html文档：



输入正确的用户名和密码，点击登陆按钮，页面会跳转到登陆成功页面：



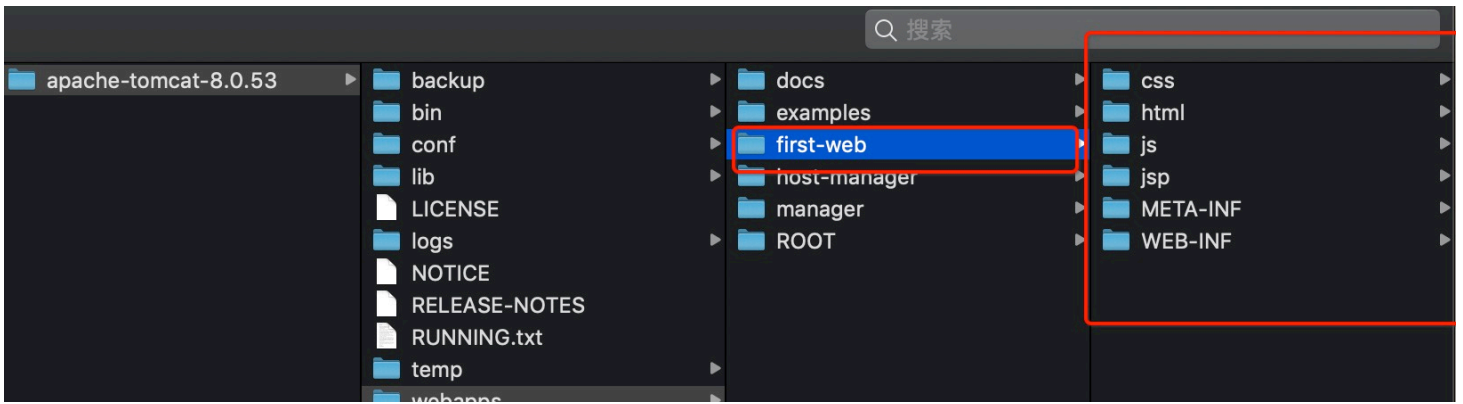
登陆成功，欢迎您，zhangsan

返回登陆页面，输入错误的用户名和密码，页面会跳转到登陆失败页面：



登陆失败，用户名密码不正确，请重新登陆！

我们打开Tomcat的webapps文件夹，会发现我们的web项目已经部署在其下：



至此，第一个Web项目部署运行完毕。这其中很多知识我们还没学习，只是为了演示效果和Web项目的目录结构。我们后续的学习，将用Maven来帮助构建Web项目。

四、Java Web应用的生命周期

Java Web的生命周期由Servlet容器来控制，主要包含三个阶段：

- 启动阶段: 该阶段主要加载有关Web应用的相关数据，创建ServletContext对象，对Filter和Servlet进行初始化。
- 运行阶段: 为客户端提供服务。
- 终止阶段: 释放Web应用所占用的资源。