

---

# 实验 10 Spring MVC 入门程序

## 一、 Spring MVC 入门

**需求：**通过一个简单的入门程序，演示 Spring MVC 的使用。该程序要求在浏览器发起请求，由 Spring MVC 接收请求并响应。

具体要求：

1. 创建一个 maven web 项目工程，命名为 trainingweb
2. 配置 pom.xml，加入 spring mvc 的依赖等
3. 创建的 Spring MVC 的配置文件名称为 spring-mvc.xml;
4. controller 层创建的包名为 com.xhcg.controller;
5. 创建 webapp 目录，用于存放 web 配置和其他页面等信息。Web 配置：创建 WEB-INF 目录及在其下创建 web.xml 文件。创建页面信息：创建 pages 目录及在其下创建 success.jsp 文件;

## 二、 环境准备（工具及版本）

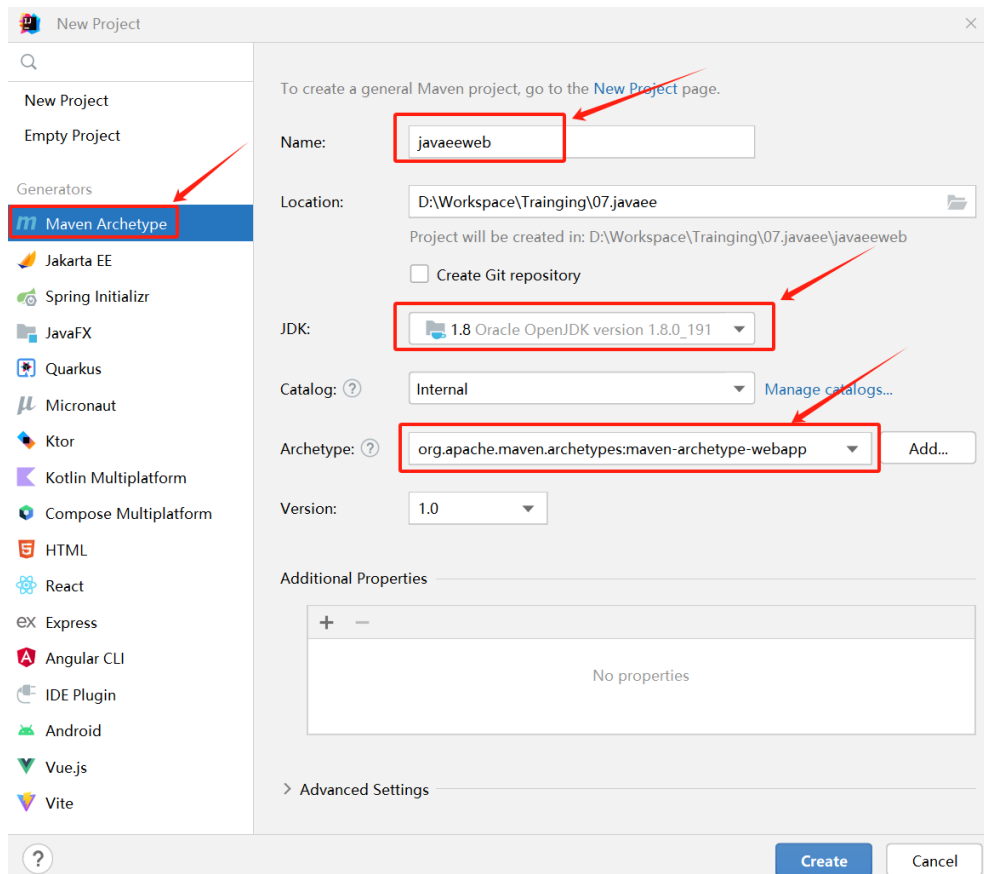
- 1、Java 版本：JDK 1.8
- 2、构建工具：Maven 3.6.3
- 3、中间件服务器：Tomcat 9.0.44**
- 4、浏览器：Google Chrome 100+以上**
- 5、开发工具：IntelliJ IDEA Ultimate 2022.3.3
- 6、数据库：MySQL 5.7+ 以上
- 7、数据库工具：Navicat 10+以上

### 三、 开发准备

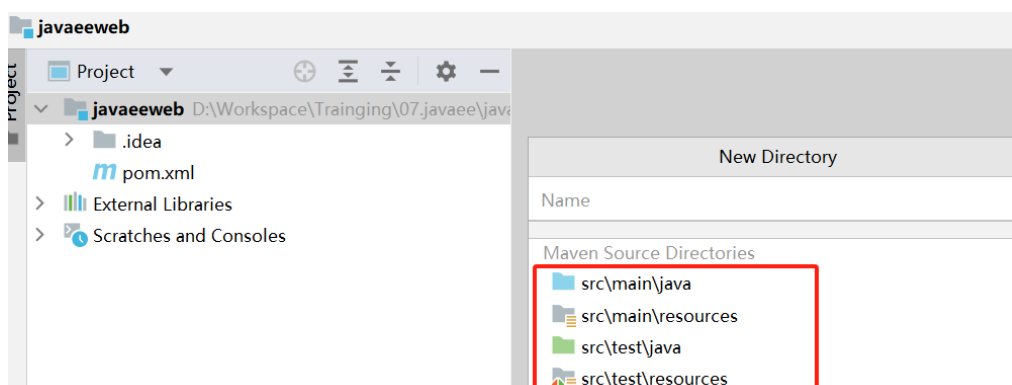
#### 1、创建 Maven Web 项目

方式一：直接创建（具体过程省略，只列出关键步骤）

IDEA 2022 创建新项目时（其他版本根据实际），选择创建 **Maven Archetype** 项目类型即可，如下



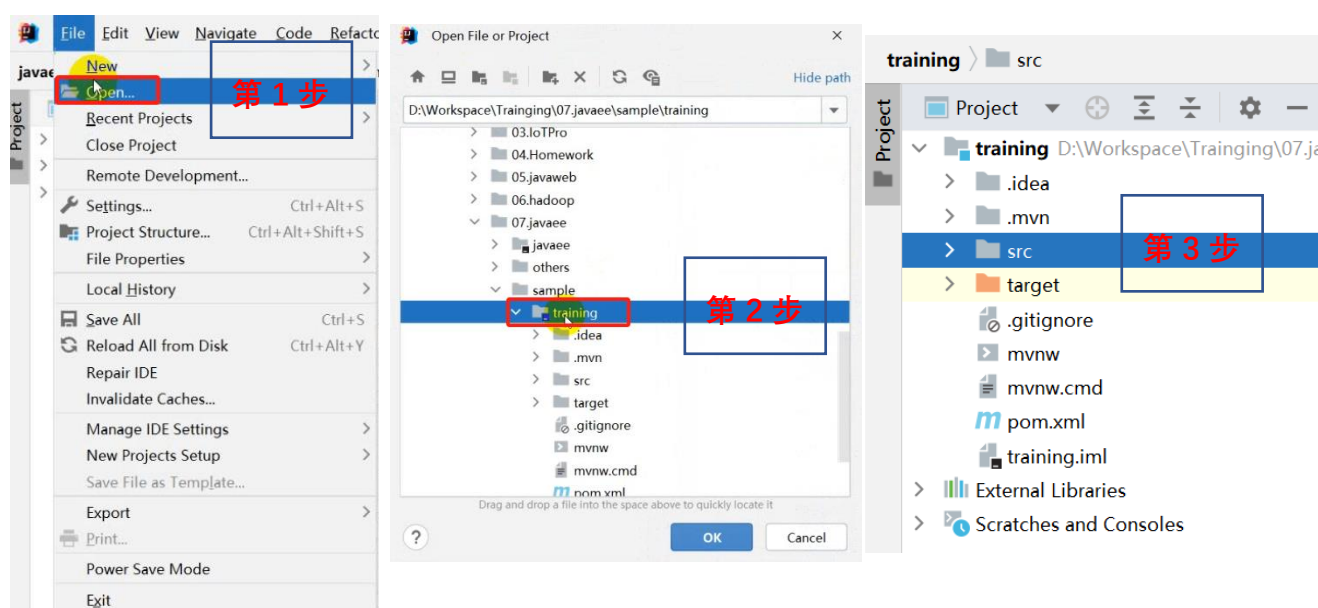
注意，上一步创建后，需要等待项目的自动构建。最终会自动创建 webapp 目录以及相应的 web 相关的配置文件。其他目录需要手工创建目录，如下：



## 方式二：导入 Maven Web 项目（推荐）

**Step1**，下载并解压 trainingweb 工程。从【学习通】→【章节】→【第 10 章】→【上机实验 10】中找到 trainingweb.zip 压缩包，下载到本地，并解压成 trainingweb 目录。

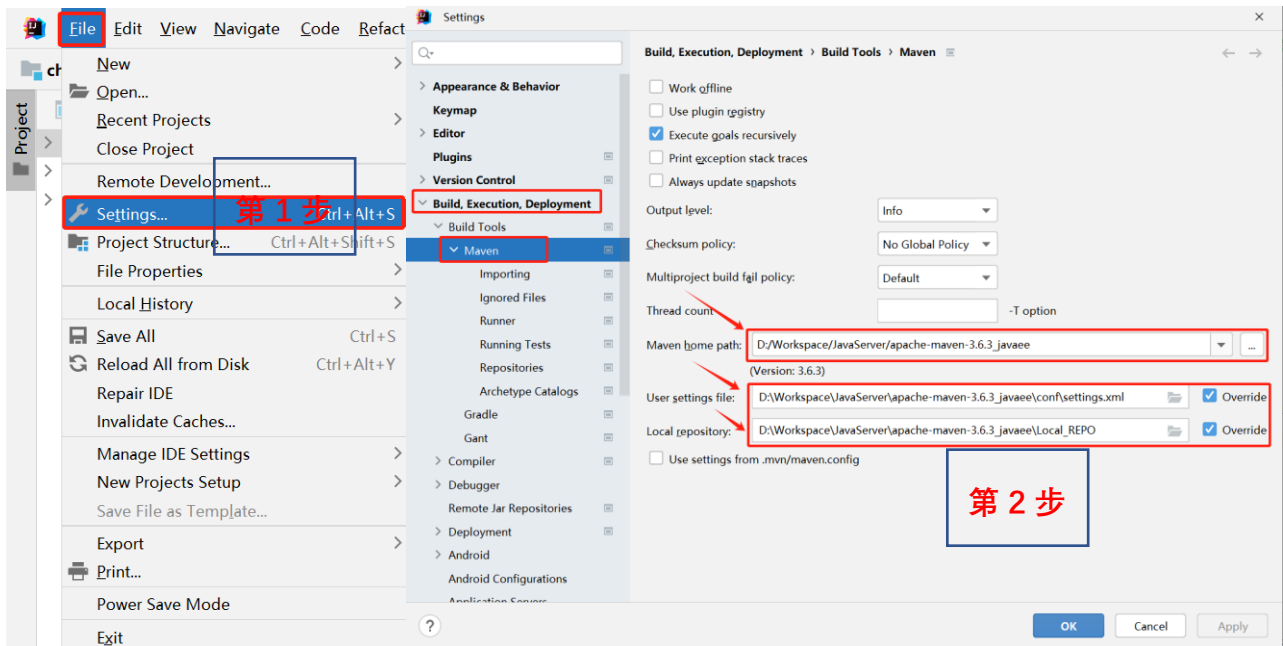
**Step2**，导入 trainingweb 工程。打开 IDEA 工具，点击【File】→【Open...】→选中上一步解压的 trainingweb 工程，如下图所示。。确定即可导入。



注意，导入工程后，对于 IDEA 是 2023 的版本，要点击左上角边的 文件夹小图标，即可查看到工程中的文件。

## 2、配置 Maven 构建工具

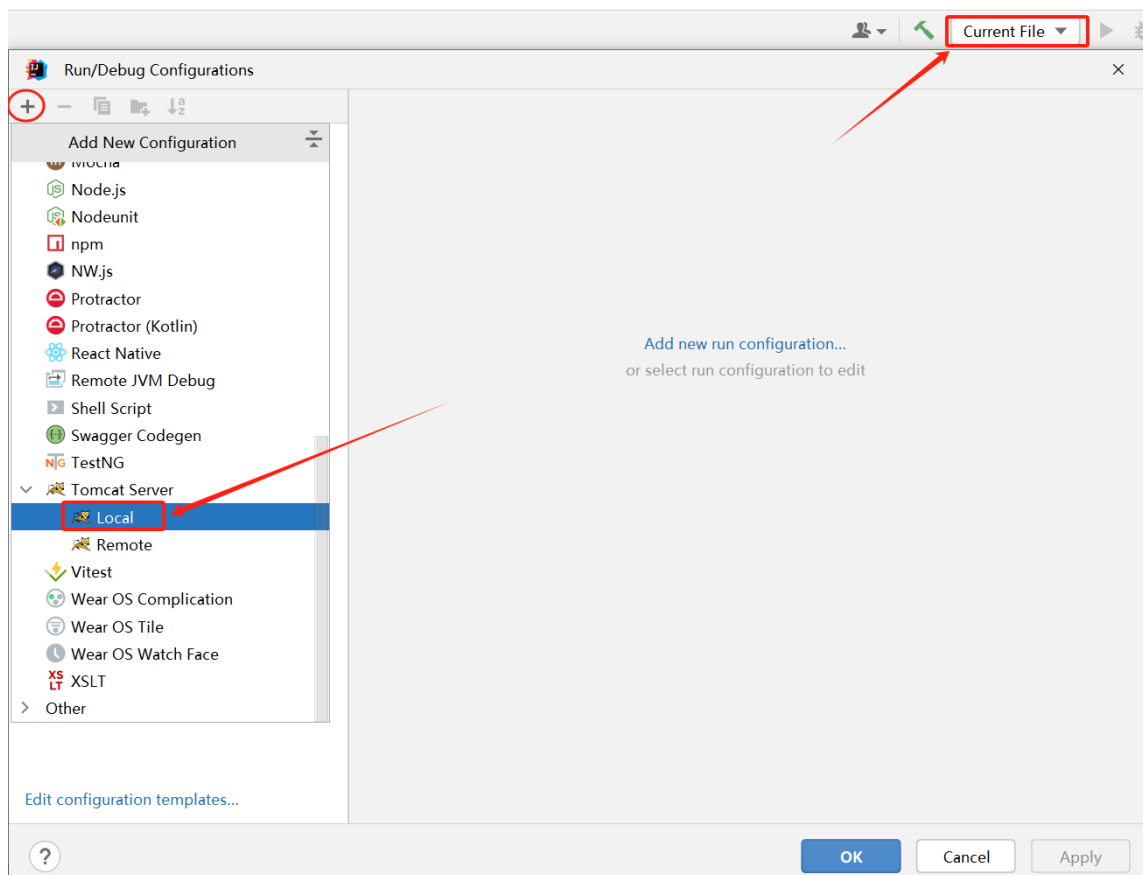
选择【File】→【Setting...】然后找到菜单单位【Build, Execution, Deployment】→【Build Tools】→【Maven】，按如下图所示，将你的本地 Maven 路径和配置文件设置好即可。



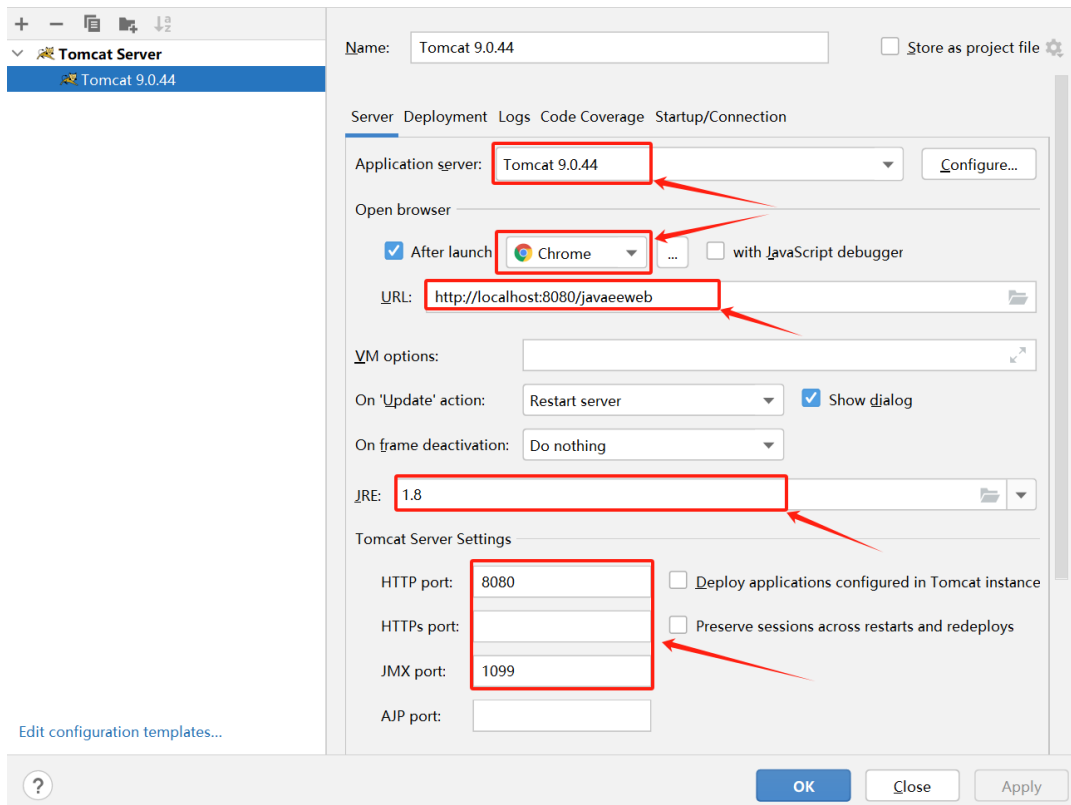
### 3、配置 Tomcat 服务器

选择【Edit Configuration】→ 点击【+】按钮，拉到 Tomcat Server，选择→【Local】，

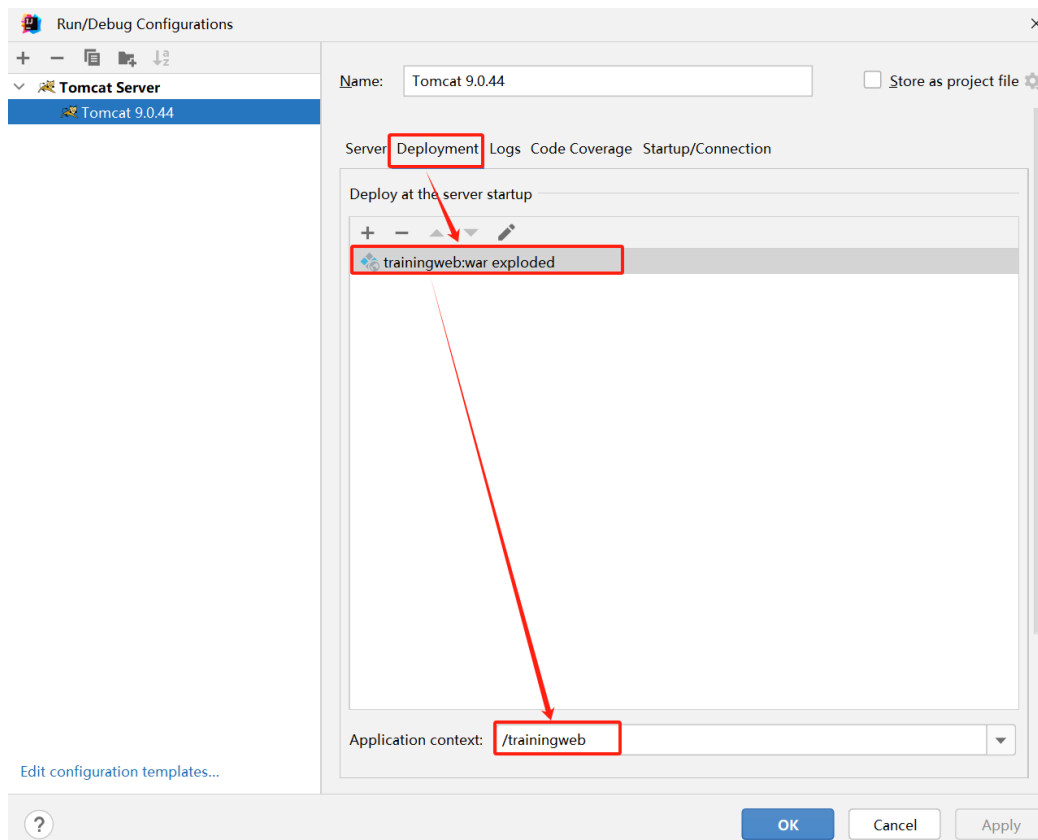
如下图所示。



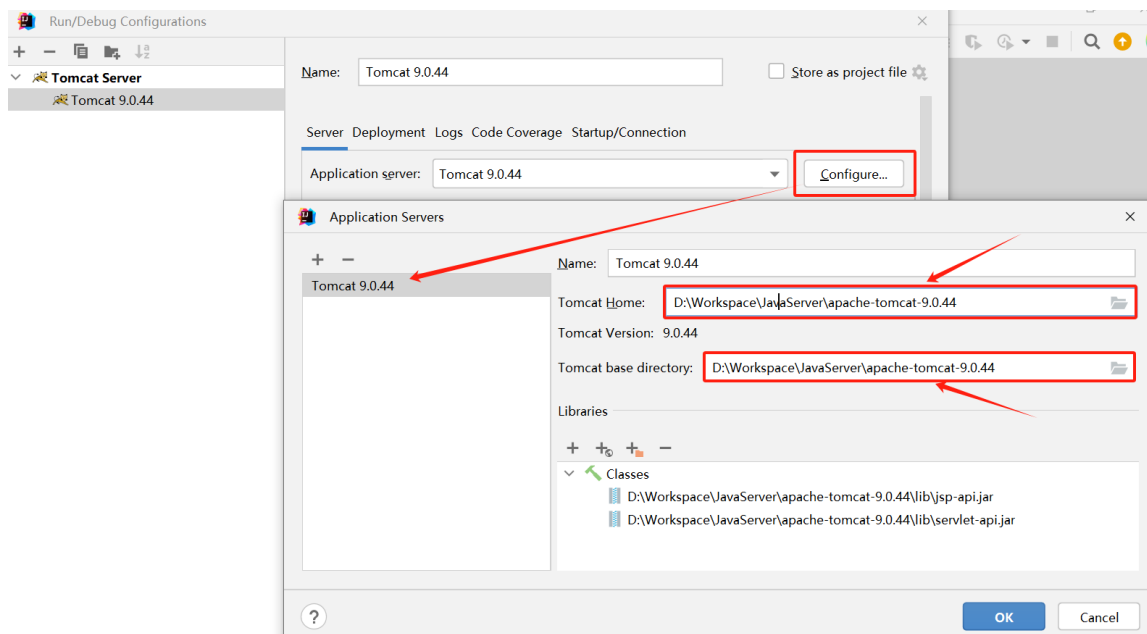
Step1: 在 Server 页签配置 Tomcat 服务器，配置如下



Step2: 切换到 Deployment 页签，设置部署包为 trainingweb:war exploded，配置如下



**Step3:** 其中 Tomcat 服务器需要指向真实的 Tomcat 目录（指向你本地的 Tomcat），配置如下。



## 四、 程序开发

### 第 1 步：配置 pom.xml

由于本章是在前 6、7、8 章的基础上学习 Spring 框架，因此，需要补充增加如下依赖，才能正常运行实验代码。pom.xml 文件需要新增依赖如下截图：

---

```
<!--Spring 核心类-->
<dependency>
  <groupId>org.springframework</groupId>
  <artifactId>spring-context</artifactId>
  <version>5.2.8.RELEASE</version>
</dependency>
<!--Spring MVC-->
<dependency>
  <groupId>org.springframework</groupId>
  <artifactId>spring-webmvc</artifactId>
  <version>5.2.8.RELEASE</version>
</dependency>
<!-- servlet -->
<dependency>
  <groupId>javax.servlet</groupId>
  <artifactId>javax.servlet-api</artifactId>
  <version>3.1.0</version>
  <scope>provided</scope>
</dependency>
<!--JSP-->
<dependency>
  <groupId>javax.servlet.jsp</groupId>
  <artifactId>jsp-api</artifactId>
  <version>2.1</version>
  <scope>provided</scope>
</dependency>
```

需要添加的依赖原配置为:

```
<dependency>
  <groupId>org.springframework</groupId>
  <artifactId>spring-context</artifactId>
  <version>5.2.8.RELEASE</version>
</dependency>
<dependency>
  <groupId>org.springframework</groupId>
  <artifactId>spring-webmvc</artifactId>
  <version>5.2.8.RELEASE</version>
</dependency>
<dependency>
  <groupId>javax.servlet</groupId>
  <artifactId>javax.servlet-api</artifactId>
  <version>3.1.0</version>
  <scope>provided</scope>
</dependency>
<dependency>
  <groupId>javax.servlet.jsp</groupId>
  <artifactId>jsp-api</artifactId>
  <version>2.1</version>
  <scope>provided</scope>
</dependency>
```

## 第 2 步：配置前端控制器 web.xml

Spring MVC 通过前端控制器拦截客户端的请求并进行转发，因此在使用 Spring MVC 时，必须先进行配置。Spring MVC 的前端控制器就是一个 Servlet，可以在项目的 web.xml 中进行配置，代码如下所示：



## 第 3 步：配置处理器映射信息和视图解析器 Spring-mvc.xml 文件

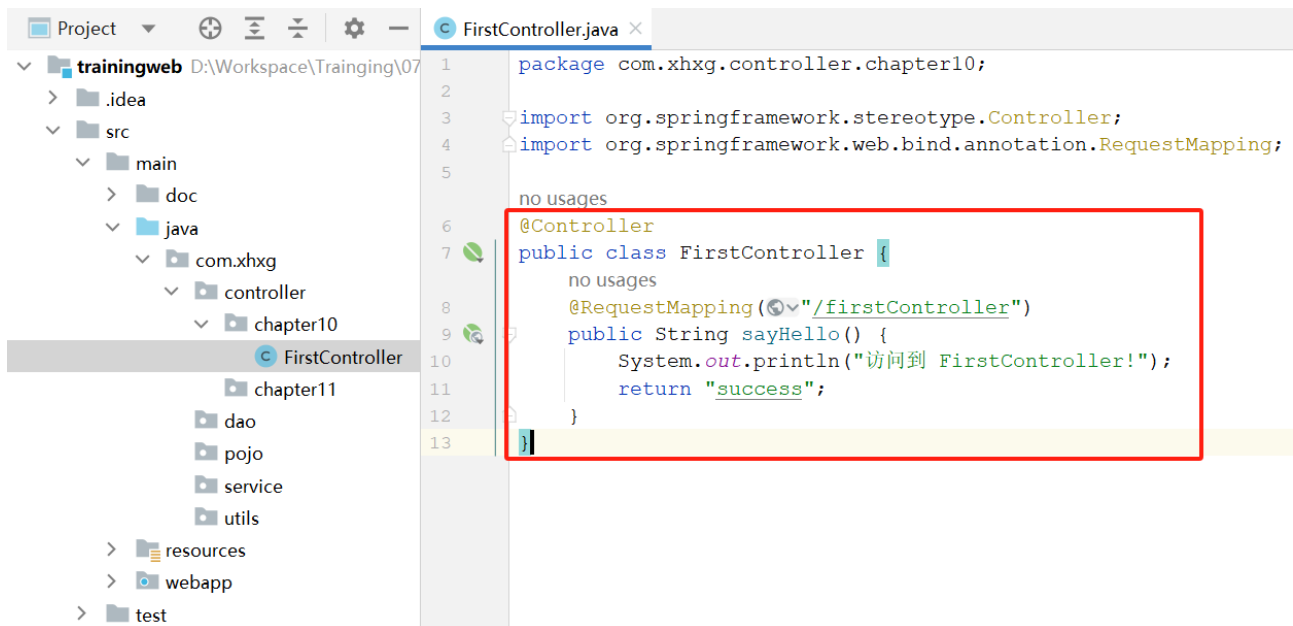
在 src/main/resources 目录下创建 spring-mvc.xml 文件，用于配置处理器映射信息和视图解析器。具体配置如下所示：



## 第 4 步：、创建处理器 Controller 类

在项目 src/main/java 目录下创建代码包：com.xhxc.controller.chapter10，创建 FirstController 类，并使用@Controller 注解注册此类为处理器类。代码如下所示：

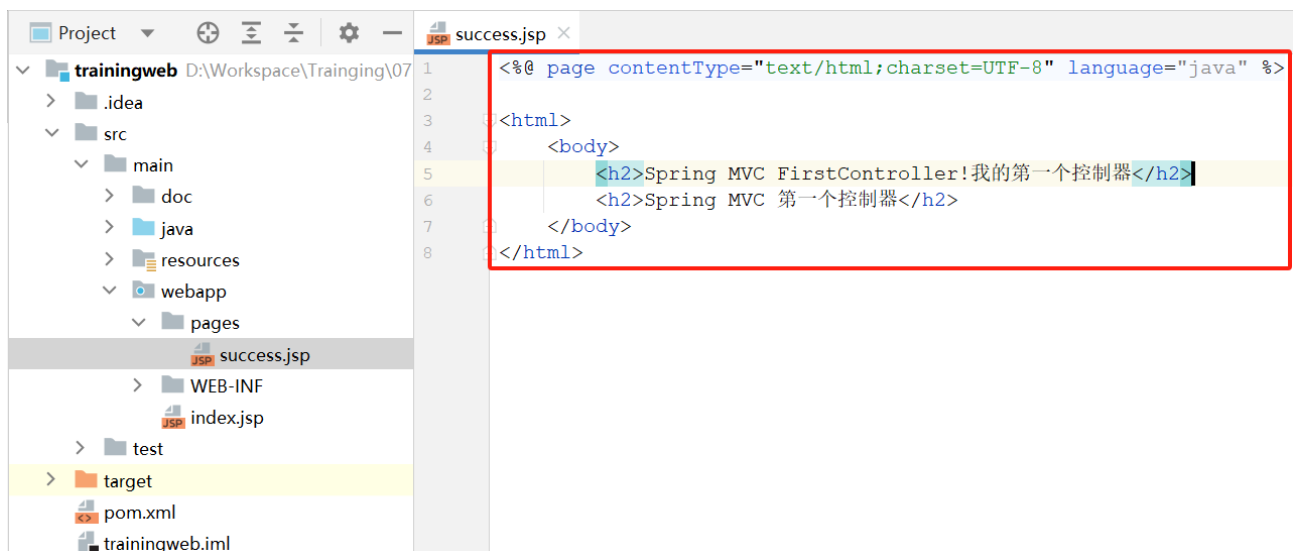




以上配置了处理器类 sayHello() 方法的访问映射地址: /firstController, 即在浏览器中通过此地址即可访问此方法。另外, sayHello() 方法也配置了访问此方法后的跳转的页面, 即 success。结合 spring-mvc.xml 配置文件中配置的视图解析器设定的前缀和后缀拼接, 即可确定视图的最终返回路径。

## 第 5 步: 创建视图 (View) 页面

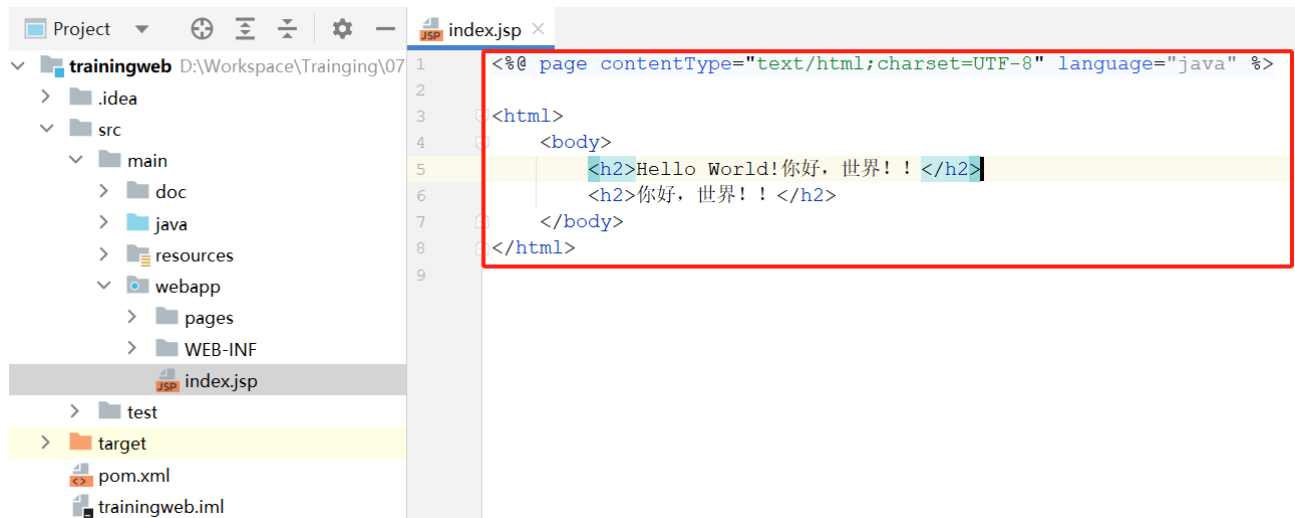
在 src/main/webapp 目录下, 创建名称为 pages 的目录, 并创建名称为 success.jsp 的页面, 用于对客户端请求进行处理后的视图进行展示。该页面内容如下:



## 第 6 步: 创建系统首页面

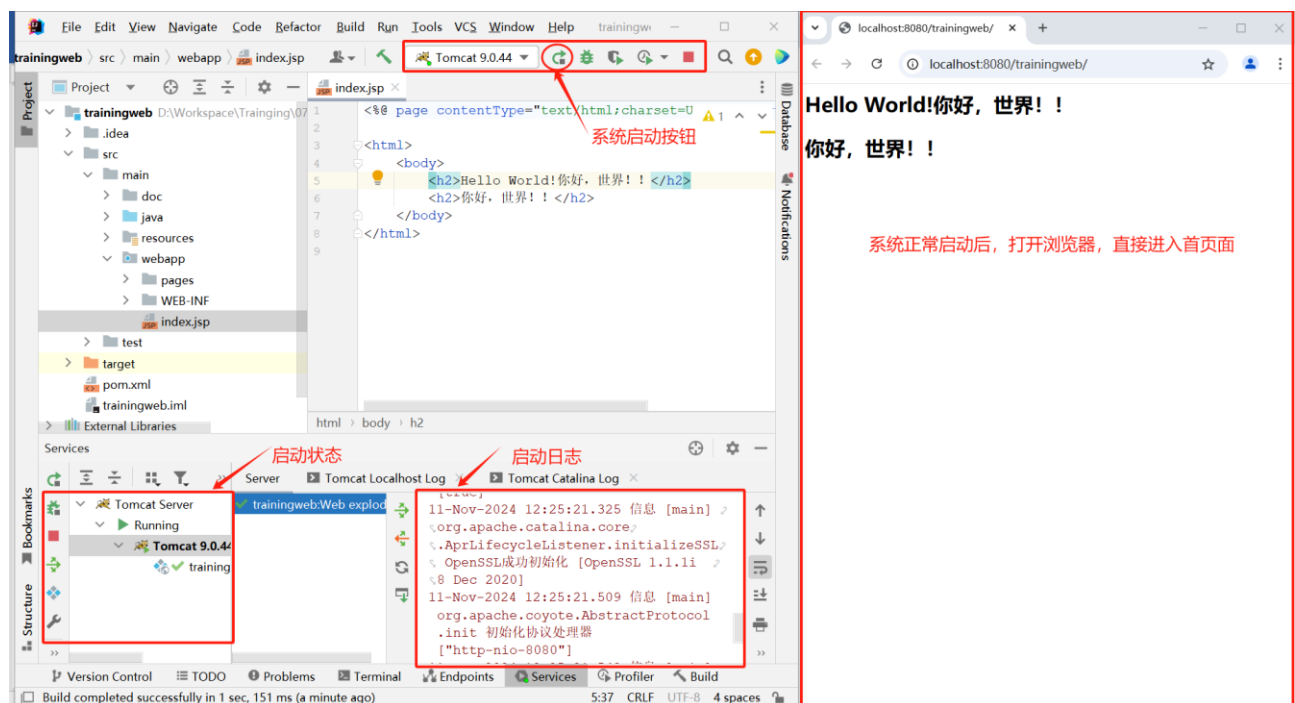
为了让系统运行更加友好, 系统启动后可以弹出一个首页, 以便测试系统运行良好。在 src/main/webapp 目录下创建名称为 index.jsp 或 default.jsp 的页面。系统启动默认会搜索这

样名称开头的页面，如有就会自动执行这个页面，作为系统的首页。该页面内容如下：



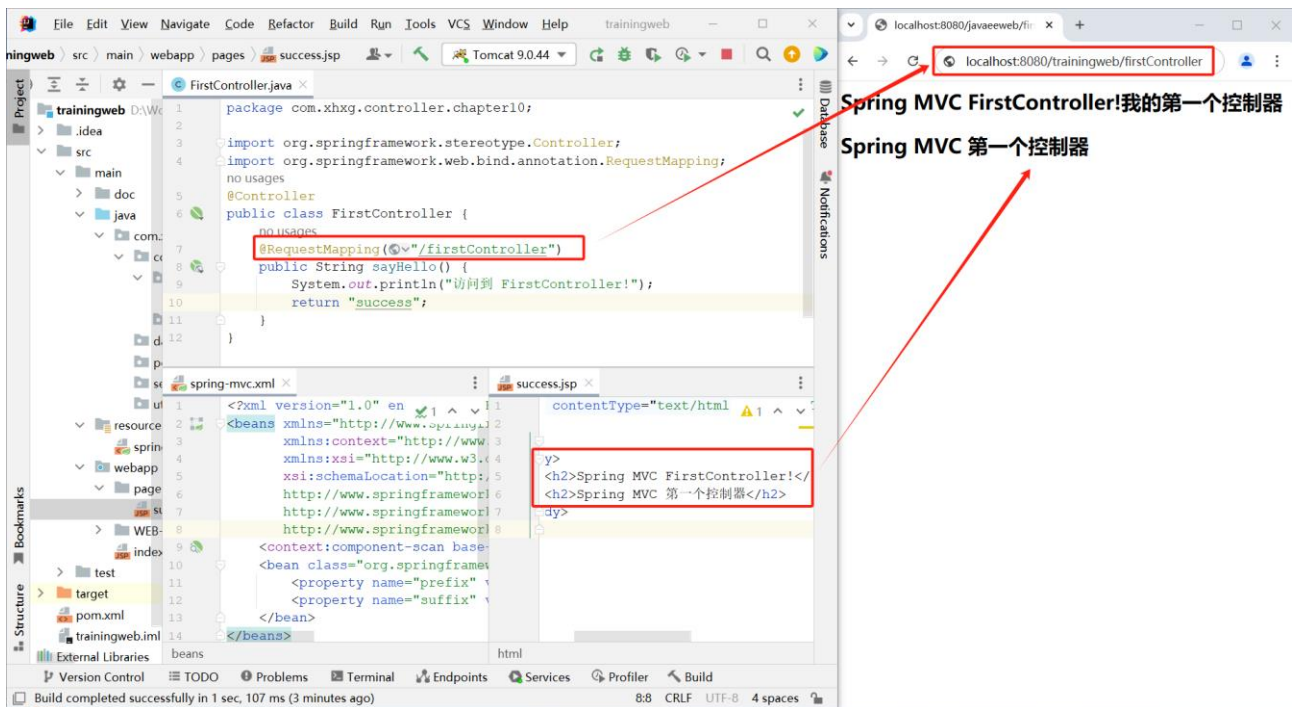
## 五、 系统运行

【第三章节】第 3 小节中已经配置好了 Tomcat 服务器（如没有配置好，请继续按第 3 小节的步骤配置好 Tomcat 服务器），所以我们直接启动此服务器即完成系统运行。



上面启动后进入的系统首页，但按照实验要求，需要执行我们在 Controller 中定义的处理程序，然后跳转到我们映射的页面中。即执行在 FirstController 中定义的访问路径：

firstController，根据 spring-mvc.xml 中的配置，执行成功会跳转到 success.jsp 页面。即效果如下：



## 六、 常见问题

待定。。