# Maven依赖

<!-- spring核心 -->

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-core</artifactId>

<version>${spring.version}</version>

</dependency>

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-beans</artifactId>

<version>${spring.version}</version>

</dependency>

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-context</artifactId>

<version>${spring.version}</version>

</dependency>

<!-- Java web -->

<dependency>

<groupId>javax.servlet</groupId>

<artifactId>javax.servlet-api</artifactId>

<version>${servlet-api.version}</version>

<scope>provided</scope>

</dependency>

<!-- spring mvc -->

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-web</artifactId>

<version>${spring.version}</version>

</dependency>

<dependency>

<groupId>org.springframework</groupId>

<artifactId>spring-webmvc</artifactId>

<version>${spring.version}</version>

</dependency>

# web.xml配置、spring配置、spring mvc配置

[..\src\main\webapp\WEB-INF\web.xml](../src/main/webapp/WEB-INF/web.xml)

（1）加载spring配置文件，启动spring

[..\src\main\webapp\WEB-INF\applicationContext.xml](../src/main/webapp/WEB-INF/applicationContext.xml)

（2）配置spring mvc的servlet、加载spring mvc配置文件，启动spring mvc

[..\src\main\webapp\WEB-INF\dispatcher-servlet.xml](../src/main/webapp/WEB-INF/dispatcher-servlet.xml)

配置文件名格式为：<servlet-name>-servlet.xml

1)扫描所有Controller

2)配置视图解析器

# Controller类

[..\src\main\java\com\zh\controller\FirstController.java](../src/main/java/com/zh/controller/FirstController.java)

（1）使用@Controller注解，spring mvc扫描读取Controller

（2）使用@RequestMapping("url")注解，映射url请求

（3）返回ModelAndView，绑定视图、传输数据

（4）使用@RequestParam("param")注解，获取请求参数

（5）spring mvc自动获取请求参数