# SpringMVC 创建Hellowold项目

## 一

首先在pom.xml上导入需要的包

|  |
| --- |
| <properties>  <spring.version>4.0.0.RELEASE</spring.version>  </properties>  <dependencies>  <dependency>  <groupId>junit</groupId>  <artifactId>junit</artifactId>  <version>3.8.1</version>  <scope>test</scope>  </dependency>  <dependency>  <groupId>org.aspectj</groupId>  <artifactId>aspectjweaver</artifactId>  <version>1.8.0</version>  </dependency>  <dependency>  <groupId>org.springframework</groupId>  <artifactId>spring-core</artifactId>  <version>${spring.version}</version>  </dependency>  <dependency>  <groupId>org.springframework</groupId>  <artifactId>spring-aop</artifactId>  <version>${spring.version}</version>  <type>jar</type>  <scope>compile</scope>  </dependency>  <dependency>  <groupId>org.springframework</groupId>  <artifactId>spring-aspects</artifactId>  <version>${spring.version}</version>  <type>jar</type>  <scope>compile</scope>  </dependency>  <dependency>  <groupId>org.springframework</groupId>  <artifactId>spring-beans</artifactId>  <version>${spring.version}</version>  <type>jar</type>  <scope>compile</scope>  </dependency>  <dependency>  <groupId>org.springframework</groupId>  <artifactId>spring-context</artifactId>  <version>${spring.version}</version>  <type>jar</type>  <scope>compile</scope>  </dependency>  <dependency>  <groupId>org.springframework</groupId>  <artifactId>spring-context-support</artifactId>  <version>${spring.version}</version>  <type>jar</type>  <scope>compile</scope>  </dependency>  <dependency>  <groupId>org.springframework</groupId>  <artifactId>spring-core</artifactId>  <version>${spring.version}</version>  <type>jar</type>  <scope>compile</scope>  </dependency>  <dependency>  <groupId>org.springframework</groupId>  <artifactId>spring-expression</artifactId>  <version>${spring.version}</version>  <type>jar</type>  <scope>compile</scope>  </dependency>  <dependency>  <groupId>org.springframework</groupId>  <artifactId>spring-jdbc</artifactId>  <version>${spring.version}</version>  <type>jar</type>  <scope>compile</scope>  </dependency>  <dependency>  <groupId>org.springframework</groupId>  <artifactId>spring-jms</artifactId>  <version>${spring.version}</version>  <type>jar</type>  <scope>compile</scope>  </dependency>  <dependency>  <groupId>org.springframework</groupId>  <artifactId>spring-orm</artifactId>  <version>${spring.version}</version>  <type>jar</type>  <scope>compile</scope>  </dependency>  <dependency>  <groupId>org.springframework</groupId>  <artifactId>spring-oxm</artifactId>  <version>${spring.version}</version>  <type>jar</type>  <scope>compile</scope>  </dependency>  <dependency>  <groupId>org.springframework</groupId>  <artifactId>spring-tx</artifactId>  <version>${spring.version}</version>  <type>jar</type>  <scope>compile</scope>  </dependency>  <dependency>  <groupId>org.springframework</groupId>  <artifactId>spring-web</artifactId>  <version>${spring.version}</version>  <type>jar</type>  <scope>compile</scope>  </dependency>  <dependency>  <groupId>org.springframework</groupId>  <artifactId>spring-webmvc</artifactId>  <version>${spring.version}</version>  <type>jar</type>  <scope>compile</scope>  </dependency>  <dependency>  <groupId>org.springframework</groupId>  <artifactId>spring-test</artifactId>  <version>${spring.version}</version>  <type>jar</type>  <scope>compile</scope>  </dependency>  <dependency>  <groupId>commons-logging</groupId>  <artifactId>commons-logging</artifactId>  <version>1.1.1</version>  </dependency>  <dependency>  <groupId>jexcelapi</groupId>  <artifactId>jxl</artifactId>  <version>2.6</version>  </dependency>  <dependency>  <groupId>javax.servlet.jsp</groupId>  <artifactId>jsp-api</artifactId>  <version>2.1</version>  <scope>provided</scope>  </dependency>  </dependencies>  <build>  <finalName>SpingDemo</finalName>  </build> |

## 在web.xml文件上加入DispatcherServlet位置

|  |
| --- |
| <web-app version=*"3.0"* xmlns=*"http://java.sun.com/xml/ns/javaee"*  xmlns:xsi=*"http://www.w3.org/2001/XMLSchema-instance"*  xsi:schemaLocation=*"http://java.sun.com/xml/ns/javaee http://java.sun.com/xml/ns/javaee/web-app\_3\_0.xsd"*>  <display-name>springMVC</display-name>  <!-- 部署 DispatcherServlet -->  <servlet>  <servlet-name>springMVC</servlet-name>  <servlet-class>org.springframework.web.servlet.DispatcherServlet</servlet-class>  <!-- 配置DispatcherServlet 的一个初始化参数，配置SpringMVC配置文件的位置和名称 -->  <init-param>  <param-name>contextConfigLocation</param-name>  <param-value>classpath:springmvc-servlet.xml</param-value>  </init-param>  <!-- 表示容器再启动时立即加载servlet -->  <load-on-startup>1</load-on-startup>  </servlet>  <servlet-mapping>  <servlet-name>springMVC</servlet-name>  <url-pattern>/</url-pattern>  </servlet-mapping>  </web-app> |

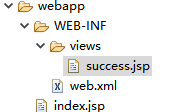
## 配置springmvc-servler.xml

|  |
| --- |
| <?xml version=*"1.0"* encoding=*"UTF-8"*?>  <beans xmlns=*"http://www.springframework.org/schema/beans"*  xmlns:xsi=*"http://www.w3.org/2001/XMLSchema-instance"*  xmlns:mvc=*"http://www.springframework.org/schema/mvc"*  xmlns:p=*"http://www.springframework.org/schema/p"*  xmlns:context=*"http://www.springframework.org/schema/context"*  xsi:schemaLocation=*"*  *http://www.springframework.org/schema/beans*  *http://www.springframework.org/schema/beans/spring-beans.xsd*  *http://www.springframework.org/schema/context*  *http://www.springframework.org/schema/context/spring-context.xsd*  *http://www.springframework.org/schema/mvc*  *http://www.springframework.org/schema/mvc/spring-mvc.xsd"*>  <!--配置自定义扫描包 -->  <context:component-scan  base-package=*"com.link.sping.controller"*></context:component-scan>  <!-- 配置视图解析器 ：如何把controller方法返回值解析为实际的物理视图 -->  <bean id=*"jspViewResolver"*  class=*"org.springframework.web.servlet.view.InternalResourceViewResolver"*>  <property name=*"prefix"* value=*"/WEB-INF/views/"* />  <property name=*"suffix"* value=*".jsp"* />  </bean>  </beans> |

## 创建一个controller对象

|  |
| --- |
| @Controller  **public** **class** Hellowold {  /\*  \* 1、通过@RequestMapping 的注解来映射请求的url  \* 2、放回值会通过视图解析器为实际的物理视图，对于InternalResourceViewResolver 视图解析器， 会做如下的解析：  \* 通过prefix +returnVal +后缀这样的方式得到实际的物理视图，然后做转发操作  \* //WEB-INF/views/success.jsp  \*/  @RequestMapping("/helloworld.do")  // @ResponseBody  **public** String hello() {  System.***out***.println("helloworld");  **return** "success";  }  } |

配置index.Jsp 和success.jsp页面



Index.jsp输入

|  |
| --- |
| <body>  <a href=*"helloworld.do"*> 登录</a>  </body> |

success.jsp输入

|  |
| --- |
| <body>  <a>success page</a>  </body> |

在网址上输入<http://localhost:8080/SpingDemo/helloworld.do>

显示

