

01 SpringMVC 入门

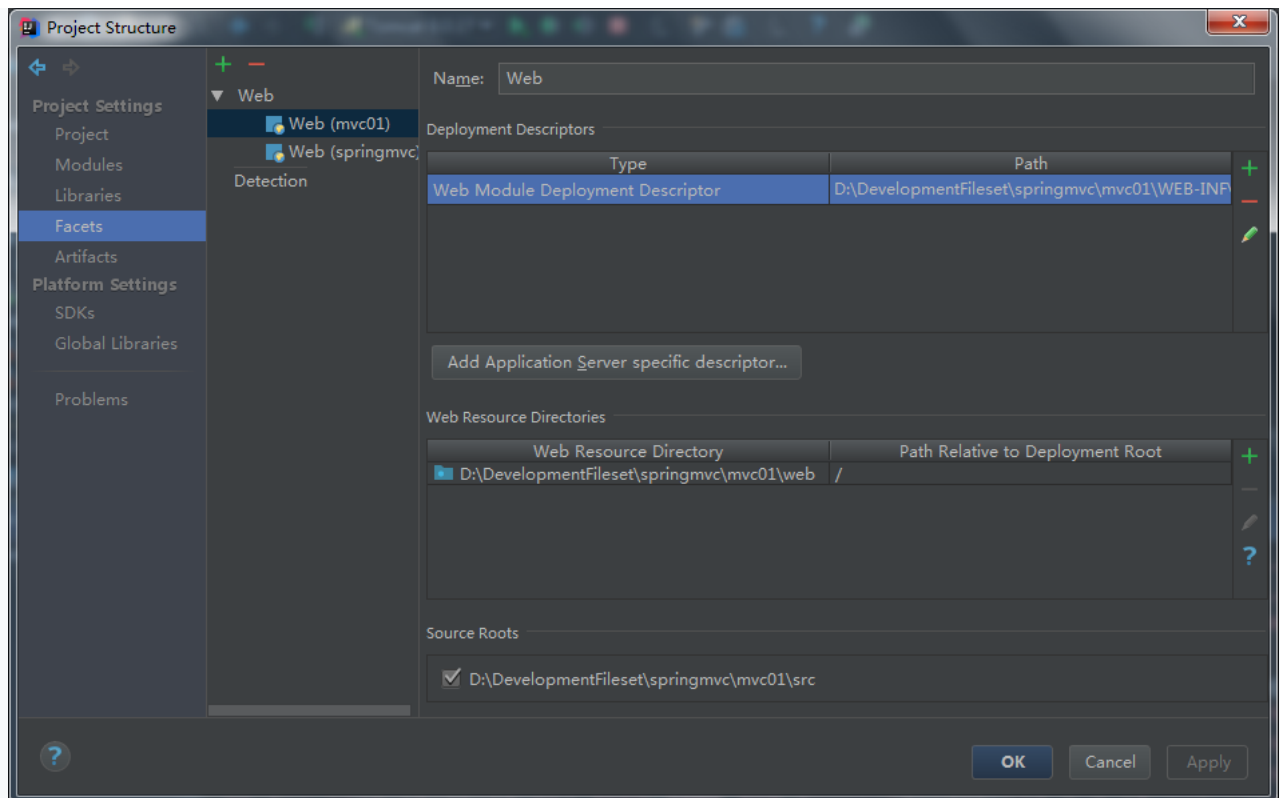
SpringMVC

一：SpringMVC简介

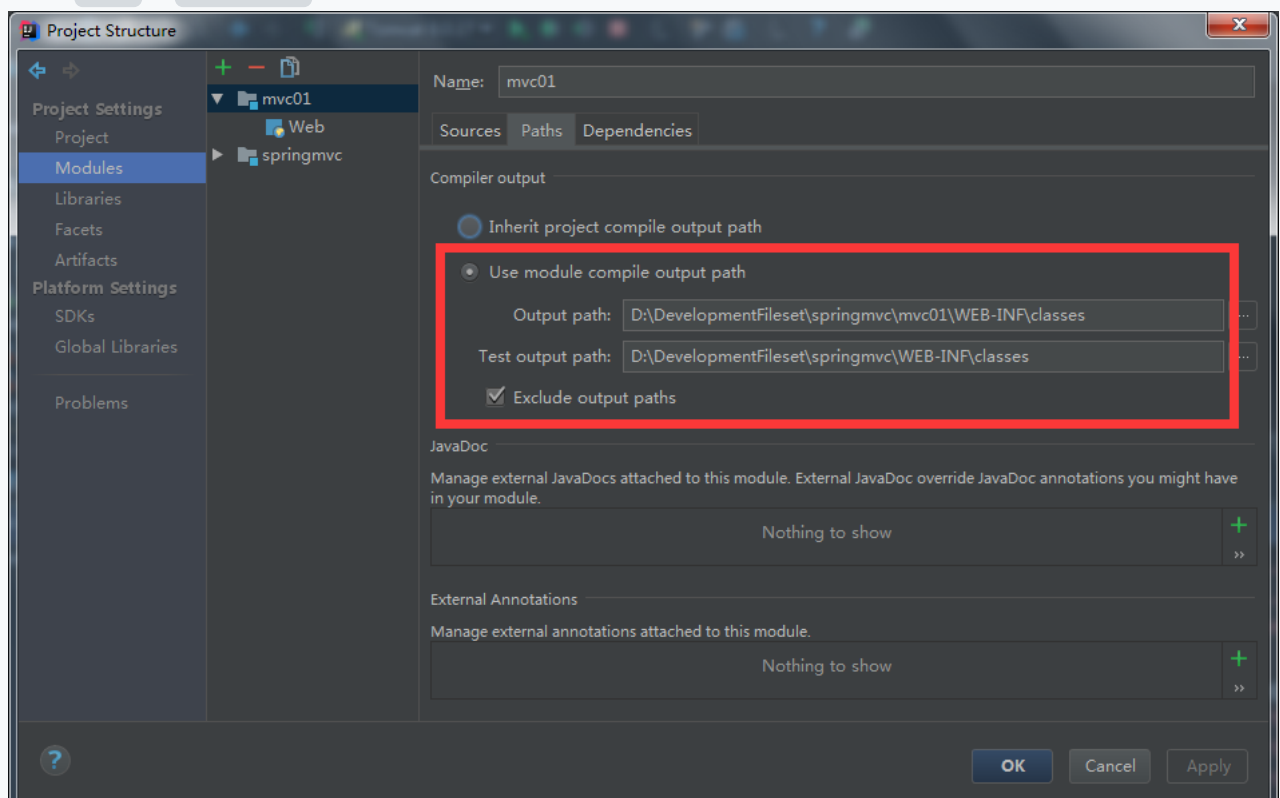
- 1、Spring Web MVC是基于ServletAPI构建的原始Web框架，从一开始就包含在Spring Framework中
- 2、正式名称"Spring Web MVC"来自其源模块 (spring-webmvc) 的名称，但它通常被称为"Spring MVC"。

二:入门案例

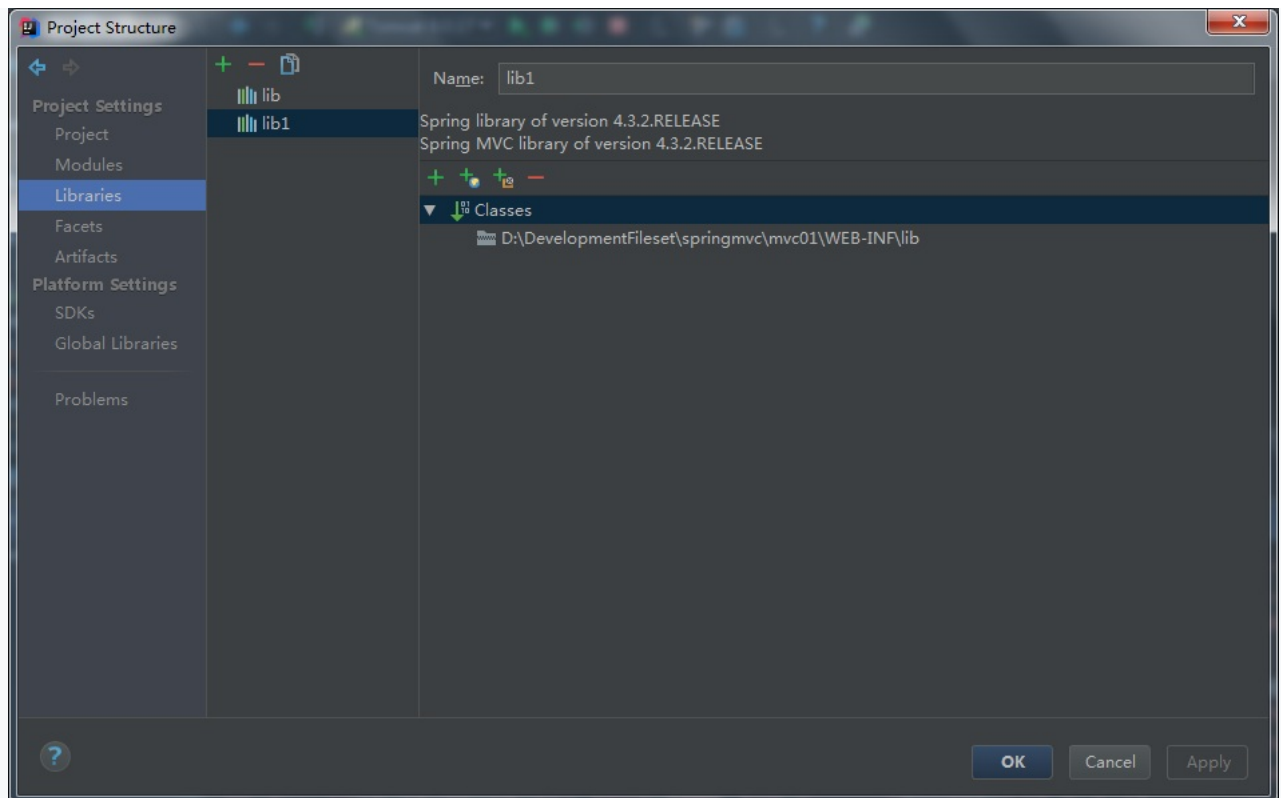
- 1、创建一个动态web工程
- 2、使用IDEA创建web工程的时候，需要在 `WEB-INF` 进行创建 `lib` 和 `classes` 文件夹，并指定路径
引入web.xml文件



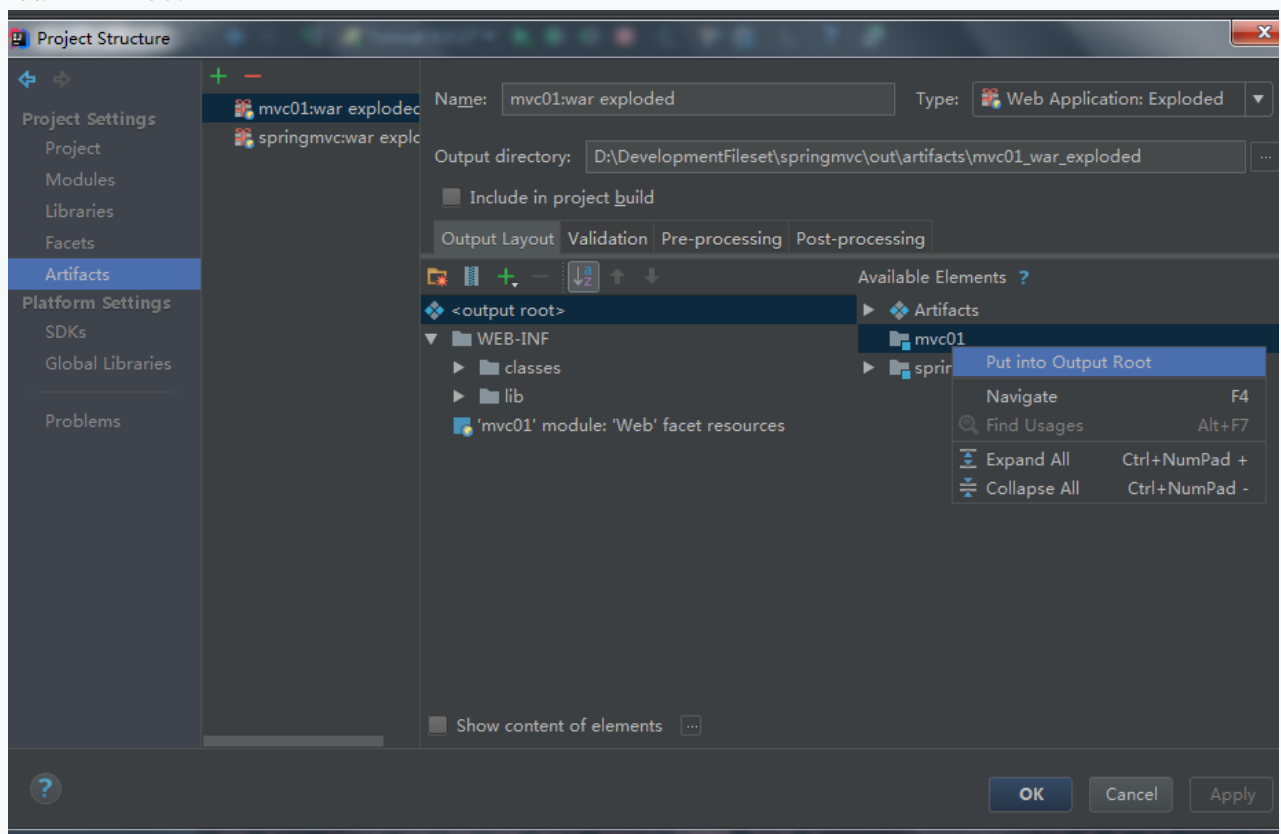
创建 lib 和 classes 文件夹



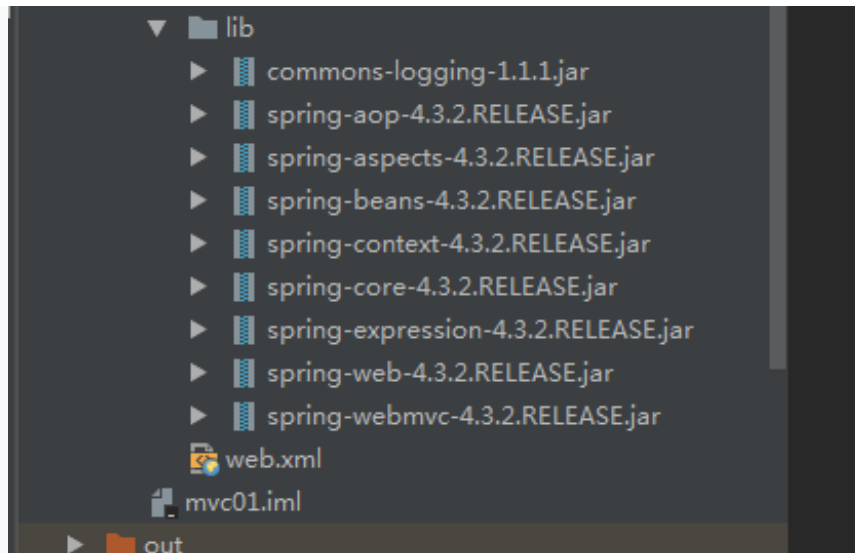
指定classes文件夹



指定lib文件夹



3、导入对应的jar包在lib目录下



1、xml文件的编写

1、web.xml文件的编写

```
1.     <servlet>
2.         <servlet-name>dispatcherServlet</servlet-name>
3.         <servlet-
4. class>org.springframework.web.servlet.DispatcherServlet</servlet-class
5. >
6.         <!-- 配置 DispatcherServlet 的一个初始化参数：配置 SpringMVC 配置文
7. 件的位置和名称 -->
8.         <!--
9.             实际上也可以不通过 contextConfigLocation 来配置 SpringMVC 的配置
10. 文件，而使用默认的。
11.             默认的配置文件的： /WEB-INF/<servlet-name>-servlet.xml
12.         -->
13.         <!--      -->
14.         <init-param>
15.             <param-name>contextConfigLocation</param-name>
16.             <param-value>classpath:springmvc.xml</param-value>
17.         </init-param>
18.         <load-on-startup>1</load-on-startup>
19.     </servlet>
20.
21.     <servlet-mapping>
22.         <servlet-name>dispatcherServlet</servlet-name>
```

```
20.         <url-pattern>/</url-pattern>
21.     </servlet-mapping>
```

2、springmvc.xml文件的编写，进行创建springMVC的xml文件

```
1.         <!-- 配置自定扫描的包 -->
2.         <context:component-scan base-package="com.spring.mvc.controller"/>
3.         <!--开启注解配置-->
4.         <mvc:annotation-driven/>
5.         <!-- 配置视图解析器：如何把 handler 方法返回值解析为实际的物理视图 -->
6.         <bean
7.             class="org.springframework.web.servlet.view.InternalResourceViewResolver"
8.             >
9.             <property name="prefix" value="/"></property>
10.            <property name="suffix" value=".jsp"></property>
11.        </bean>
```





2、controller层的编写

1、创建一个请求类

```
1.     @Controller
2.     public class TestMvc01 {
3.         @RequestMapping("/springmvc")
4.         public String Test01() {
5.             System.out.println("ssssssssssssss");
6.             return "success";
7.         }
8.     }
```

2、请求<http://localhost:8080/springmvc/>

 |  http://localhost:8080/springmvc/

★收藏 ▾  网址导航  游戏中心  小说大全  爱淘宝

asdasdasdasdasdasdasd爱上茶水的