

Evan

Only let oneself become strong enough, good enough, can afford the life that you want to.

☰ 目录视图

☰ 摘要视图

🔒 订阅专栏

从创业到再就业，浅述对程序员职业生涯的看法 征文 | 你会为 AI 转型么？ 赠书：7月大咖新书机器学习/Android/python

Spring--环境配置

标签：spring 环境配置

2016-05-18 19:37 120人阅读 评论(0) 收藏 举报

☰ 分类： Spring (1) web后台 (12)

目录

1.1 Spring jar包下载

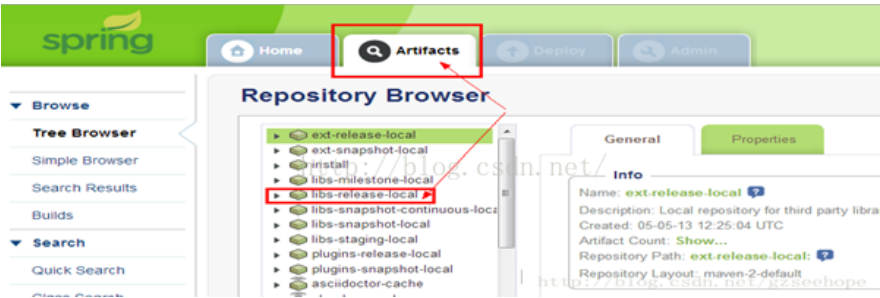
1.2 Hello World

参考资料

1.1 Spring jar包下载

(1) 进入官网<http://repo.spring.io> (或者 <http://maven.springframework.org/release/org/springframework/spring/>)

(2) 在资源库界面点击Artifacts标签，然后点击libs-release-local，展开后依次点击org -> springframework -> spring。



然后，就可以看到发布的各个版本，选一个自己需要的版本，点击Download。

(我这里下载的是当前最新版本spring-framework-4.1.6.RELEASE-dist)



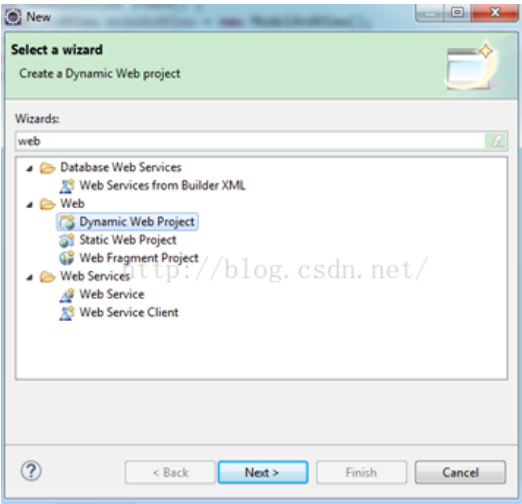
1.2 Hello World

SpringMVC 配置过程中很容易出现各式各样的问题，很是费劲。

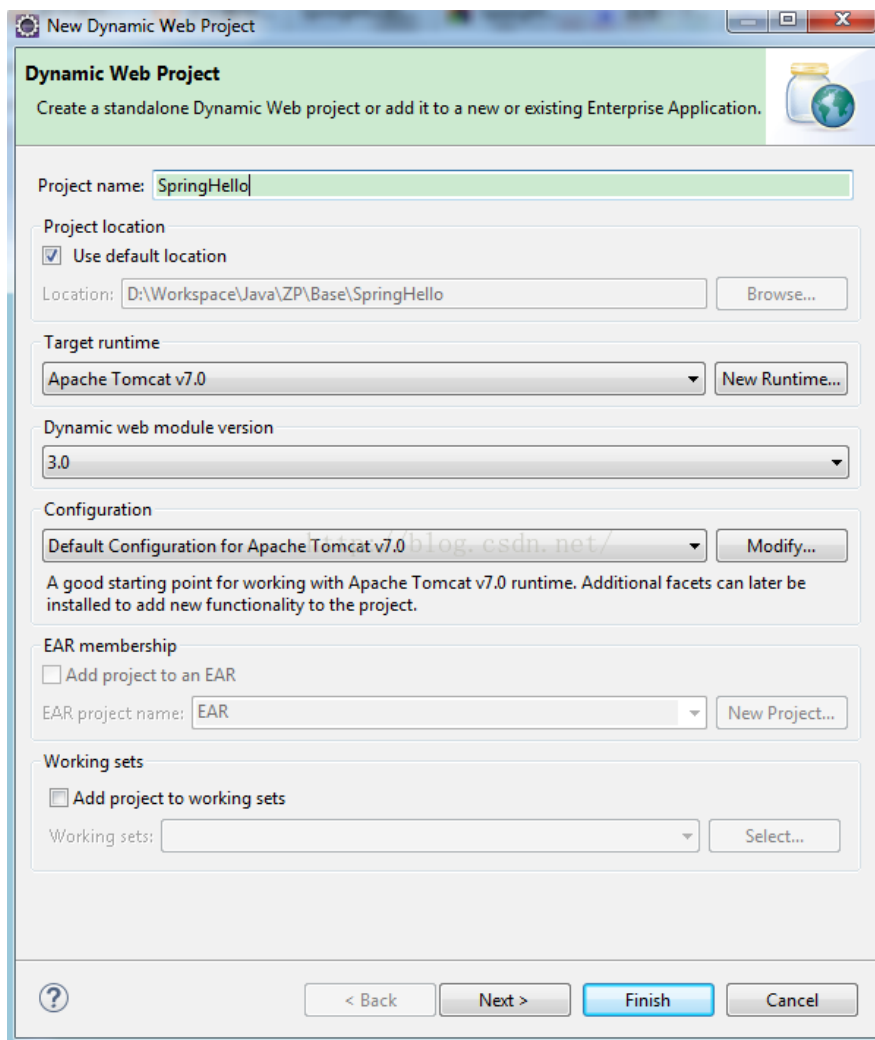
下面提供傻瓜式创建HelloWorld工程的方式，希望大家少走弯路。

(1) 创建一个Dynamic Web Project

点击File -> New -> Other, 输入web， 可以找到Dynamic Web Project。



新建一个名为SpringHello的工程。可参考下图的配置 。



(2) 导入spring的jar包

在WEB-INF目录下新建一个lib目录（如果没有lib目录）。

将下载的spring-framework-4.1.6.RELEASE-dist\spring-framework-4.1.6.RELEASE\libs中的jar包拷贝到WEB-INF\lib 目录下

为图省事，本人将所有jar都拷贝进来了。

此外，spring框架还依赖一个外部jar包：commons-logging-1.2.jar，需要下载并导入。

注：顺便提一下，spring-framework-4.1.6.RELEASE-dist\spring-framework-4.1.6.RELEASE\libs下的jar包分为普通jar包、sources jar包和javadoc jar包。sources 和javadoc分别包含了普通jar包的源码和java文档。

有兴趣想看spring源码的朋友可以导入，导入方法网上有很多介绍，这里不赘述。

(3) 添加web.xml

web.xml文件中声明了servlet的清单。

```
[html]
01. <span style="font-size:14px;"><?xml version="1.0" encoding="UTF-8"?>
02. <web-app version="2.5" xmlns="http://java.sun.com/xml/ns/javaee"
03.     xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
04.     xsi:schemaLocation="
05.         http://java.sun.com/xml/ns/javaee
06.         http://java.sun.com/xml/ns/javaee/web-app_2_5.xsd">
07.
08.
09.     <display-name>SpringHello</display-name>
10.
11.     <!-- 声明前端控制器 -->
12.     <servlet>
13.         <servlet-name>spring</servlet-name>
14.         <servlet-class>org.springframework.web.servlet.DispatcherServlet</servlet-class>
15.         <init-param>
16.             <!-- 如果不设置的话，默认的servlet配置文件名为xxx-servlet.xml -->
17.             <param-name>contextConfigLocation</param-name>
18.             <param-value>/WEB-INF/spring.xml</param-value>
19.         </init-param>
20.         <load-on-startup>1</load-on-startup>
21.     </servlet>
22.
23.     <!-- 声明DispatcherServlet处理哪些URL -->
24.     <servlet-mapping>
25.         <servlet-name>spring</servlet-name>
26.         <url-pattern>/</url-pattern><!-- 映射到/表示会处理所有的请求 -->
27.     </servlet-mapping>
28. </web-app>
29. </span>
```

SpringMVC的核心是DispatcherServlet，这个servlet的角色是前端控制器。所有的servlet都是在web.xml中配置的，DispatcherServlet当然也不例外。

如果不指定contextConfigLocation，默认情况下servlet配置文件的名字形式应该为servlet名-servlet.xml，这个servlet名就是<servlet-name>标签中的名字。

(4) 添加servlet——spring.xml

新建一个名为spring.xml的文件（这个文件名必须和web.xml中的servlet对应上），内容如下：

```
[html]
01. <span style="font-size:14px;"><?xml version="1.0" encoding="UTF-8"?>
02. <beans xmlns="http://www.springframework.org/schema/beans"
```

```

03.     xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
04.     xmlns:p="http://www.springframework.org/schema/p"
05.     xmlns:context="http://www.springframework.org/schema/context"
06.     xmlns:util="http://www.springframework.org/schema/util"
07.     xmlns:mvc="http://www.springframework.org/schema/mvc"
08.     xsi:schemaLocation="
09.         http://www.springframework.org/schema/beans
10.         http://www.springframework.org/schema/beans/spring-beans-3.0.xsd
11.         http://www.springframework.org/schema/context
12.         http://www.springframework.org/schema/context/spring-context-3.0.xsd
13.         http://www.springframework.org/schema/util
14.         http://www.springframework.org/schema/util/spring-util-3.0.xsd
15.         http://www.springframework.org/schema/mvc
16.         http://www.springframework.org/schema/mvc/spring-mvc-3.0.xsd">
17.
18.     <!-- 默认的注解映射的支持 -->
19.     <mvc:annotation-driven />
20.
21.     <!-- 如果当前请求为"/"时，则转发到"/login/index" -->
22.     <mvc:view-controller path="/" view-name="forward:/login/index" />
23.
24.     <!-- 设置后，将查找使用@Component(@Controller是它的一种具体化)注解的类并将其注册为Bean -->
25.     <context:component-scan base-package="com.demo.web.controllers" />
26.
27.     <!-- 配置视图解析器（该配置的是jstl解析器的一个扩展解析类） -->
28.     <bean
29.         class="org.springframework.web.servlet.view.InternalResourceViewResolver">
30.         <property name="prefix" value="/WEB-INF/view/" />
31.         <property name="suffix" value=".jsp" />
32.     </bean>
33. </beans></span>

```

(5) 添加Controller——LoginController.java

Controller是SpringMVC的控制器，用于处理页面请求。

新建一个名为LoginController的Java文件，内容如下：

```

[java]
01. <span style="font-size: 14px;">package com.demo.web.controllers;
02.
03. import org.springframework.stereotype.Controller;
04. import org.springframework.web.bind.annotation.RequestMapping;
05. import org.springframework.web.bind.annotation.RequestMethod;
06. import org.springframework.web.servlet.ModelAndView;
07.
08. @Controller
09. public class LoginController {
10.     @RequestMapping(value="/login/index", method = RequestMethod.GET)
11.     public ModelAndView index() {
12.         ModelAndView mav = new ModelAndView();
13.         mav.addObject("message", "Hello World! First program!");
14.         mav.setViewName("index"); // 设置视图名称
15.         return mav;
16.     }
17. }</span>

```

Spring2.5版本引入了注解方式，所以，我们可以很方便的使用@Controller来将一个类标注为控制器。

在`spring.xml`中，我们定义了`<context:component-scan>`标签，这样`LoginController`会自动被发现并注册为Bean。需要注意的是，`spring.xml`应放在`WEB-INF`目录下。

`@RequestMapping`可以将一个URL映射到一个实体类或一个特殊的handler方法上。`RequestMethod.GET`表示只接受get请求。

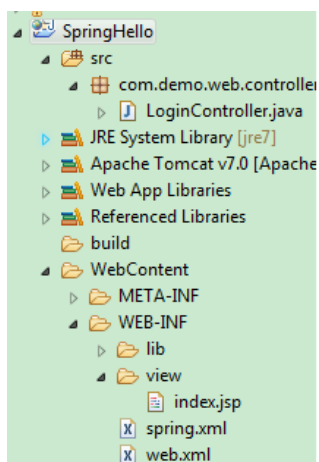
(6)添加视图——index.jsp

在`/WEB-INF`文件夹下新建一个`view`文件夹，并添加一个名为`index`的jsp文件，内容如下：

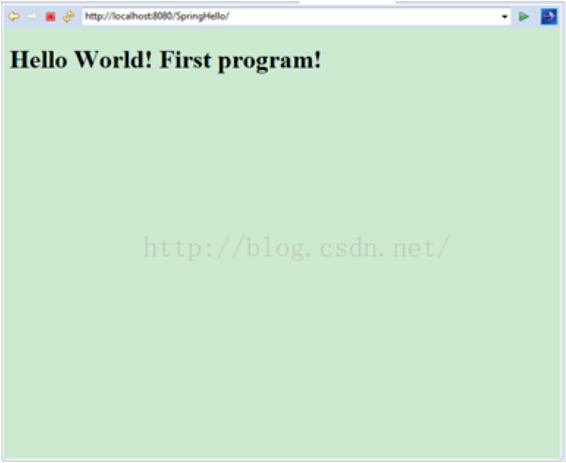
```
[html]
01. <span style="font-size:14px;">
02. <%@ page language="java" contentType="text/html; charset=ISO-8859-1"
03.     pageEncoding="ISO-8859-1"%>
04. <!DOCTYPE html PUBLIC "-//W3C//DTD HTML 4.01 Transitional//EN" "http://www.w3.org/TR/html4/loose.dtd">
05. <html>
06. <head>
07. <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
08. <title>Spring Hello World</title>
09. </head>
10. <body>
11.     <h1>${message}</h1>
12. </body>
13. </html></span>
```

(7) 运行项目

经过上面的步骤，项目文件结构如下



运行项目，选择Run As -> Run On Server结果如下:



参考资料

Spring实战（第3版）

<http://www.cnblogs.com/liukemng/p/3725582.html>

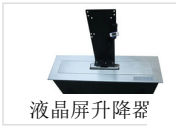
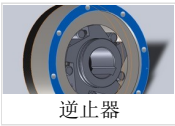
from: <http://www.cnblogs.com/jingmoxukong/p/4503462.html>

顶 0 踩 0

- [上一篇](#) Sencha Touch2 -- 11.1：定义具有关联关系的模型
- [下一篇](#) Eclipse构建Maven项目

相关文章推荐

- spring boot之多环境配置
- ibatis+spring+struts 环境配置
- Spring flex 搭建服务 之 编程环境配置和我的第一...
- 手把手struts2.0+hibernate3.1+spring2.0环境配...
- Spring PropertyPlaceholderConfigurer 自定义...
- Activiti环境配置、项目搭建、与Spring集成、简...
- jdk+tomcat+myeclipse+spring开发环境配置小记
- Sshs环境配置：
- spring boot 多环境配置开发及打包
- 用spring profile或maven profile实现多环境配置...



猜你在找

- 机器学习之概率与统计推断
- 机器学习之凸优化
- 响应式布局全新探索
- 深度学习基础与TensorFlow实践
- 前端开发在线峰会

- 机器学习之数学基础
- 机器学习之矩阵
- 探究Linux的总线、设备、驱动模型
- 深度学习之神经网络原理与实战技巧
- TensorFlow实战进阶：手把手教你做图像识别应用