J2ee项目从0搭建进阶(一):springmvc 集成 tiles3实现页面模板

孤天浪雨 2016-08-28 09:32:55 3570 收藏

展开

1、背景

在我们实际的web项目开发中，会有很多页面，同样每个页面都需要有header、body、footer等，有时候这些header内容都是一样的，所以我们就需要想办法重用它们，这就有了我们的需求，也就引出了tiles来实现模板的概念。

这边转载了两篇文章可以看一下有助了解：

Apache Tiles的基本使用

[页面模板框架对比] Apache Tiles VS Sitemesh

2、pom依赖

<!-- tiles start -->

<dependency>

<groupId>org.apache.tiles</groupId>

<artifactId>tiles-extras</artifactId>

<version>3.0.7</version>

</dependency>

<dependency>

<groupId>org.apache.tiles</groupId>

<artifactId>tiles-servlet</artifactId>

<version>3.0.7</version>

</dependency>

<dependency>

<groupId>org.apache.tiles</groupId>

<artifactId>tiles-jsp</artifactId>

<version>3.0.7</version>

</dependency>

<!-- tiles end -->

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

3、配置tiles视图解析

tiles视图解析配置推荐配置在springmvc的视图解析器之前，这样就会优先匹配tiles配置来进行视图解析。（配在springmvc配置文件中）

我这边将tiles.xml与tiles.xml在同目录下。

<!-- 引入tiles配置文件 -->

<bean id="tilesConfigurer" class="org.springframework.web.servlet.view.tiles3.TilesConfigurer" >

<property name="definitions">

<list>

<value>/WEB-INF/tiles.xml</value>

</list>

</property>

</bean>

<!-- 配置视图解析器 -->

<bean id="tilesViewResolver" class="org.springframework.web.servlet.view.UrlBasedViewResolver">

<property name="viewClass">

<value>org.springframework.web.servlet.view.tiles3.TilesView</value>

</property>

</bean>

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

4、配置tiles.xml

我们先准备几个基础页面

layout目录下：header.jsp、menu.jsp、footer.jsp、template.jsp

tiles(为了演示目录随便建的)目录下：body1.jsp、body2.jsp[这两个页面用来演示页面切换]

具体页面内容大家可以见最后的github链接中的演示项目中查看，这边给出模板页template.jsp和tiles配置文件tiles.xml的内容

template.jsp：这里定义模板的样式及所要引用的页面

<%@ page contentType="text/html;charset=UTF-8" language="java" %>

<%@ taglib uri="http://tiles.apache.org/tags-tiles" prefix="tiles" %>

<table border="1" cellpadding="2" cellspacing="2" align="center">

<tr>

<td height="30" colspan="2"><tiles:insertAttribute name="header" />

</td>

</tr>

<tr>

<td height="250"><tiles:insertAttribute name="menu" /></td>

<td width="350"><tiles:insertAttribute name="body" /></td>

</tr>

<tr>

<td height="30" colspan="2"><tiles:insertAttribute name="footer" />

</td>

</tr>

</table>

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

tiles.xml：这边可以使用通配符，这样就能根据我们返回的视图名称进行自动匹配上模板展示了，这边配成\*.page就是所有以.page结尾的视图都会自动选择模板展示。

<?xml version="1.0" encoding="UTF-8"?>

<!DOCTYPE tiles-definitions PUBLIC "-//Apache Software Foundation//DTD Tiles Configuration 3.0//EN" "http://tiles.apache.org/dtds/tiles-config\_3\_0.dtd">

<tiles-definitions>

<!-- 定义一个基础模板页 -->

<definition name="template\_base" template="/WEB-INF/layout/template.jsp"></definition>

<!-- 不继承直接引用也行 -->

<definition name="\*.page" extends="template\_base">

<put-attribute name="title" value="" />

<put-attribute name="header" value="/WEB-INF/layout/header.jsp" />

<put-attribute name="menu" value="/WEB-INF/layout/menu.jsp" />

<put-attribute name="body" value="/WEB-INF/tiles/{1}.jsp" />

<put-attribute name="footer" value="/WEB-INF/layout/footer.jsp" />

</definition>

</tiles-definitions>

1

2

3

4

5

6

7

8

9

10

11

12

13

14

15

16

17

5、TilesController配置

@Controller

public class TilesController {

@RequestMapping("/body1")

public String index(Model model) {

return "body1.page";

}

@RequestMapping("/body2")

public String home(Model model) {

return "body2.page";

}

}

1

2

3

4

5

6

7

8

9

10

11

12

13

14

6、运行示例

7、github项目地址

https://github.com/gubaijin/buildmavenweb