<http://blog.csdn.net/superdog007/article/details/56488550>

配置jsp视图解析器

<!-- spring-mvc jsp视图解析器 -->

<bean id="viewResolver"

class="org.springframework.web.servlet.view.InternalResourceViewResolver">

<property name="viewClass"

value="org.springframework.web.servlet.view.JstlView"></property>

<property name="prefix" value="/WEB-INF/views/"></property>

<property name="suffix" value=".jsp"></property>

<property name="order" value="2"></property>

</bean>

配置title视图解析器

<!-- 配置tiles模板 -->

<!-- tilesResolver -->

<bean id="tilesConfigurer"

class="org.springframework.web.servlet.view.tiles3.TilesConfigurer">

<property name="definitions">

<list>

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

</list>

</property>

<property name="preparerFactoryClass"

value="org.springframework.web.servlet.view.tiles3.SpringBeanPreparerFactory" />

</bean>

<!-- tiles视图解释器 -->

<bean id="tilesViewResolver"

class="org.springframework.web.servlet.view.UrlBasedViewResolver">

<property name="order" value="1"></property>

<property name="viewClass"

value="org.springframework.web.servlet.view.tiles3.TilesView" />

</bean>

配置html解析器

<!-- html视图解析器 必须先配置freemarkerConfig,注意html是没有prefix前缀属性的-->

<bean id="freemarkerConfig" class="org.springframework.web.servlet.view.freemarker.FreeMarkerConfigurer">

<property name="templateLoaderPath">

<value>/WEB-INF/views/</value>

</property>

</bean>

<bean id="htmlviewResolver"

class="org.springframework.web.servlet.view.freemarker.FreeMarkerViewResolver">

<property name="suffix" value=".html" />

<property name="order" value="0"></property>

<property name="contentType" value="text/html;charset=UTF-8"></property>

</bean>

1.注意配置html视图解析器时，出现了404错误，仔细了解 当一次请求是需要经过两次拦截器，即html–>拦截器–>controller–>拦截器—>html，其中如果按照下面配置，第一次请求拦截器是正确的，第二次拦截就包404错误。

<bean id="viewResolver1"

class="org.springframework.web.servlet.view.InternalResourceViewResolver">

<property name="viewClass"

value="org.springframework.web.servlet.view.JstlView"></property>

<property name="prefix" value="/WEB-INF/views/"></property>

<property name="suffix" value=".html"></property>

<property name="order" value="0"></property>

</bean>

解决方法有三种，

1. 第一种，使用标题配置“html”视图解析器这样配置，个人推荐这种配置。注意这种配置html是没有前缀属性的，所以需要先配置freemarkerConfig，前缀保存在freemarkerConfig中。
2. 第二种，在web.xml中加入下面代码，表示对静态资源.html不拦截。

<servlet-mapping>

<servlet-name>default</servlet-name>

<url-pattern>\*.html</url-pattern>

</servlet-mapping>

第三种，使用

<mvc:resources location="/" mapping="/\*\*/\*.html"/>