Spring Web MVC视图

1. 上次课程回顾

@ComponentScan->@Configuration class ->basePakages->@Component class

->Bean definition->BeanDefinitionRegistry->Beans->BeanFactory->getBean or @Autowired

Bean的生命周期

实例化：Bean Class->Bean Object

初始化前：Bean before/pre init()

BeanPostProcessor#prostProcessBeanInitializtion

初始化：

InitializingBean#afterPropertiesSet()

初始化后：

BeanPostProcessor#prostProcessBeanInitializtion

销毁

DisposableBean#destroy()

1. Thymeleaf视图技术(太么礼服)

渲染上下文（模型）Model

* Spring Web MVC

接口类型

1. Model
2. ModelMap
3. ModelAndView

Model ->ModelMap

View

注解类型

@ModelAttribute

学习技巧

假设需要了解的技术是thymeleaf->thymeleaf Properties->ThymeleafProperties

找到@ConfigurationProperties确定前缀

|  |
| --- |
| @ConfigurationProperties(prefix = **"spring.thymeleaf"**) **public class** ThymeleafProperties {  …  } |

字段名与属性一一对应：

**spring.thymeleaf.cache=true**

**spring.thymeleaf.mode=HTML**

|  |
| --- |
| @ConfigurationProperties(prefix = **"spring.thymeleaf"**) **public class** ThymeleafProperties {  …  **this**.cache = **true**;  **private** String mode = **"HTML"**;  } |

EL表达式

* 字符
* 逻辑表达式

1. if else

* 迭代表达式

1. for each/while

* 反射

1. Java Reflection
2. CGLib

模板寻址：

prefix+view name+suffix

1. Spring MVC模板渲染逻辑

C：DispatcherServlet

M：

Spring MVC

* Model/ModelMap/ModelAndView
* @ModelAttribute

模板引擎

通用方式：内建/内嵌自己工具变量

* JSP内置九大对象

1. Scope 上下文（Spring Scope）

PageContext（page变量）：

关注于当前页面。A forward B->A变量只能用于A页面使用，不能共享给B；A 和B可以采用同名变量，同时使用。

Servlet Request：- WebApplicationContext. SCOPE\_REQUEST

关注于当前请求。A forward B->A请求变量可以用于B页面。

Servlet Session：- WebApplicationContext. SCOPE\_SESSION

关注于当前会话。A forward/redirect B->A请求变量可以用于B页面。

Servlet ServletContext：- WebApplicationContext. SCOPE\_APPLICATION

关注于当前应用。用户A 和用户B会话可以共享。

1. JSP内置变量

out（ServletResponse#getWriter）

exception（Throwable）

config（Servlet Config）

page（JSP Servlet）

response（ServletResponse）

* Thymeleaf 内置变量

StandardExpressionObjectFactory-> ITemplateContext 内置上下文

1. 上下文模型
2. #strings
3. #numbers
4. …

V：

* 视图对象

1. Servlet

RequestDispatcher#forward

RequestDispatcher#include

HttpServletResponse#sendRedirect

1. Spring MVC

UrlBasedViewResolver

view：

forward：InternalResourceView（forward:）

redirect：RedirectView（redirect:）

1. Struts

ForwardAction

RedirectAction

* 视图处理对象

1. Spring MVC

\*.do->DispatcherServlet->Controller->View->ViewResolver->View#render->HTML->HttpServletResponse

1. Thymeleaf ViewResolver-> ThymeleafViewResolver
2. Thymeleaf View-> ThymeleafView

通过模板名称解析模板资源

TemplateResolution

读取资源，渲染内容HTML

HTML内容输出到Response

源码路径：

* org.thymeleaf.TemplateEngine#process(org.thymeleaf.TemplateSpec,

org.thymeleaf.context.IContext, java.io.Writer)

* org.thymeleaf.engine.TemplateManager#parseAndProcess

JSP

1. ViewResolver-> InternalResourceViewResolver
2. View-> JstlView
3. Struts

\*.do->ActionServlet->Action->ForwardAction->RequstDispatcher->JSP->HTML-> HttpServletResponse

1. 国际化

Locale

Spring MVC

Locale

* Servlet->ServletRequest#getLocale

**Accept-Language:** zh-CN,zh;q=0.9

* LocaleContextHolder

1. 来自于：

DispatcherServlet

org.springframework.web.servlet.FrameworkServlet#initContextHolders

1. 存储在ThreadLocal

MessageSourceAutoConfiguration->

Spring Boot

MessageSource

* MessageSourceAutoConfiguration
* MessageSourceProperties

前缀：**spring.messages**

**messages.properties 为默认配置，当找不到语言配置时使用。**

**messages\_zh\_cn.properties 中文配置**

**messages\_en.properties 英文配置**

**如何动态改变语言配置：**

**国际化解析器：**

CookieLocaleResolver

SesssionLocaleResolver

LocaleChangeInterceptor 动态改变语言配置

Thymeleaf

#key->messageSource.get