

第十二章：Spring 国际化

小马哥 · mercyblitz



扫码试看/订阅

《小马哥讲 Spring 核心编程思想》视频课程

Spring 国际化

1. Spring 国际化使用场景
2. Spring 国际化接口
3. 层次性 MessageSource
4. Java 国际化标准实现
5. Java 文本格式化
6. MessageSource 开箱即用实现
7. MessageSource 内建依赖
8. 课外资料
9. 面试题精选



Spring 国际化使用场景

- 普通国际化文案
- Bean Validation 校验国际化文案
- Web 站点页面渲染
- Web MVC 错误消息提示

Spring 国际化接口

- 核心接口
 - `org.springframework.context.MessageSource`
- 主要概念
 - 文案模板编码 (code)
 - 文案模板参数 (args)
 - 区域 (Locale)

层次性 MessageSource

层次性优势：搜索范围变大，可以从更大范围查找，类似类加载双亲委派

- Spring 层次性接口回顾
 - org.springframework.beans.factory.HierarchicalBeanFactory
 - org.springframework.context.ApplicationContext
 - org.springframework.beans.factory.config.BeanDefinition
- Spring 层次性国际化接口
 - org.springframework.context.HierarchicalMessageSource

都是层次性的

Java 国际化标准实现

- 核心接口
 - 抽象实现 - `java.util.ResourceBundle`
 - Properties 资源实现 - `java.util.PropertyResourceBundle`
 - 例举实现 - `java.util.ListResourceBundle`

Java 国际化标准实现

- ResourceBundle 核心特性
 - Key-Value 设计
 - 层次性设计
 - 缓存设计
 - 字符编码控制 - `java.util.ResourceBundle.Control` (@since 1.6)
 - Control SPI 扩展 - `java.util.spi.ResourceBundleControlProvider` (@since 1.8)

Java 文本格式化

- 核心接口
 - `java.text.MessageFormat`
- 基本用法
 - 设置消息格式模式 - `new MessageFormat(...)`
 - 格式化 - `format(new Object[]{...})`
- 消息格式模式
 - 格式元素: `{ArgumentIndex (,FormatType,(FormatStyle))}`
 - `FormatType`: 消息格式类型, 可选项, 每种类型在 `number`、`date`、`time` 和 `choice` 类型选其一
 - `FormatStyle`: 消息格式风格, 可选项, 包括: `short`、`medium`、`long`、`full`、`integer`、`currency`、`percent`

Java 文本格式化

- 高级特性
 - 重置消息格式模式
 - 重置 `java.util.Locale`
 - 重置 `java.text.Format`

MessageSource 开箱即用实现

- 基于 ResourceBundle + MessageFormat 组合 MessageSource 实现
 - `org.springframework.context.support.ResourceBundleMessageSource`
- 可重载 Properties + MessageFormat 组合 MessageSource 实现
 - `org.springframework.context.support.ReloadableResourceBundleMessageSource`

MessageSource 内建依赖

- MessageSource 内建 Bean 可能来源
 - 预注册 Bean 名称为: “messageSource” , 类型为: MessageSource Bean
 - 默认内建实现 - DelegatingMessageSource
 - 层次性查找 MessageSource 对象

课外资料

- Spring Boot 为什么要新建 MessageSource Bean?
 - AbstractApplicationContext 的实现决定了 MessageSource 内建实现
 - Spring Boot 通过外部化配置简化 MessageSource Bean 构建
 - Spring Boot 基于 Bean Validation 校验非常普遍

面试题

沙雕面试题 - Spring 国际化接口有哪些?

答:

- 核心接口 - `MessageSource`
- 层次性接口 - `org.springframework.context.HierarchicalMessageSource`



我真的没笑

面试题

996 面试题 - Spring 有哪些 MessageSource 内建实现?



答:

- `org.springframework.context.support.ResourceBundleMessageSource`
- `org.springframework.context.support.ReloadableResourceBundleMessageSource`
- `org.springframework.context.support.StaticMessageSource`
- `org.springframework.context.support.DelegatingMessageSource`

面试题

劝退面试题 - 如何实现配置自动更新 MessageSource?



答:

- 主要技术
 - Java NIO 2: `java.nio.file.WatchService`
 - Java Concurrency : `java.util.concurrent.ExecutorService`
 - Spring: `org.springframework.context.support.AbstractMessageSource`



扫码试看/订阅

《小马哥讲 Spring 核心编程思想》视频课程