

第4章 Spring Boot整合Thymeleaf

● ————— ●
《Spring Boot企业级开发教程（第2版）》



黑马程序员
www.itheima.com

传智教育旗下
高端IT教育品牌

学习目标/Target



了解Spring Boot支持的模板引擎，能够说出Spring Boot支持的模板引擎

了解Thymeleaf简介，能够说出Thymeleaf是什么和Thymeleaf的特点

掌握Thymeleaf常用属性，能够在模板文件中正确使用Thymeleaf常用属性

掌握Thymeleaf标准表达式，能够在模板文件中正确使用Thymeleaf标准表达式

掌握图书管理案例的实现，能够使用Spring Boot整合Thymeleaf完成图书管理案例



开发Web应用时，为了让项目代码耦合性更低，可维护性更高，通常会采用MVC设计模式来实现对应的模型、视图和控制器，让业务逻辑、数据、界面显示分离。最初的Web应用的视图是由HTML元素组成的静态界面，而如今的Web应用更倾向于在界面中动态显示对应的数据。Spring Boot框架提供了一些常用视图技术的整合支持，并推荐通过整合模板引擎来实现Web应用界面的动态化展示。下面将对Spring Boot支持的模板引擎进行介绍，并对其中常用的Thymeleaf模板引擎进行整合。



4.1

注册Java Web三大组件

4.2

Spring Boot管理Spring MVC

4.3

文件上传



4.1

Spring Boot支持的模板引擎

>>> 4.1 Spring Boot支持的模板引擎



黑马程序员
www.itheima.com

传智教育旗下
高端IT教育品牌



了解Spring Boot支持的模板引擎，能够说出Spring Boot支持的模板引擎



FreeMarker

FreeMarker是一个基于模板生成输出文本的模板引擎，输出文本可以是HTML页面、电子邮件、配置文件等。

Groovy

Groovy是一种基于JVM的敏捷开发语言，它结合了Python、Ruby和Smalltalk的许多强大特性。

Thymeleaf

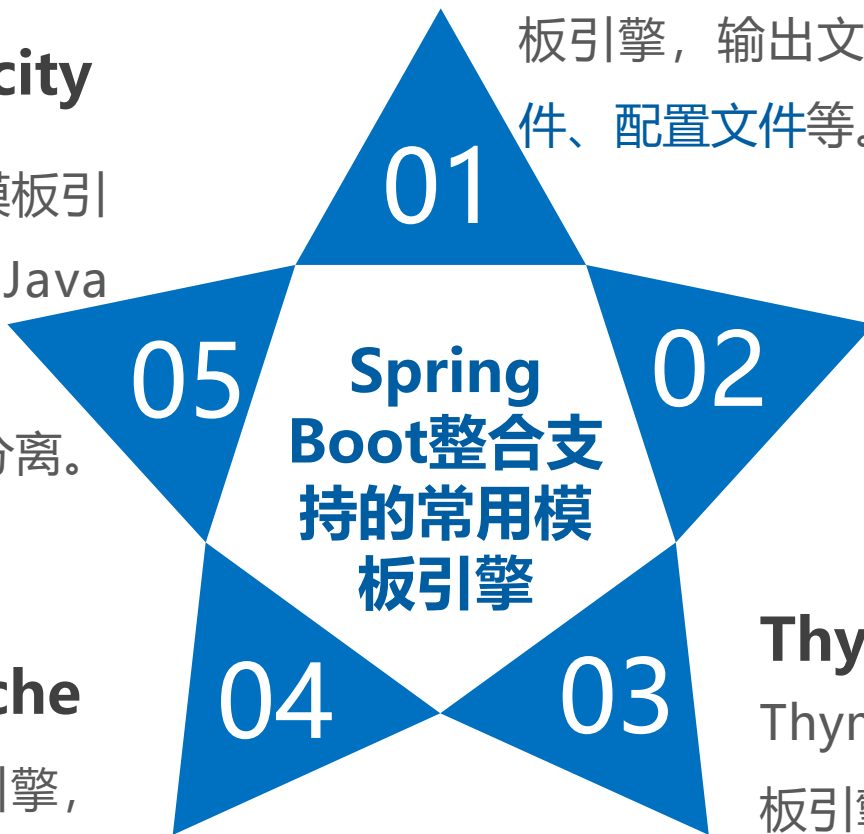
Thymeleaf是采用Java语言编写的模板引擎，可用于Web与非Web环境中的应用开发。是Spring Boot官方推荐使用的模板引擎。

Velocity

Velocity是一个基于Java的模板引擎。通过特定的语法获取在Java对象的数据，填充到模板中，从而实现界面和Java代码的分离。

Mustache

Mustache是轻逻辑的模板引擎，它是一个JavaScript模板，用于对JavaScript分离展示。





Spring Boot不太支持JSP模板，并且没有提供对应的整合配置，这是因为使用嵌入式Servlet容器的Spring Boot应用程序对JSP模板有一些限制，具体如下。

- Spring Boot默认使用嵌入式Servlet容器以JAR包方式进行项目打包部署，这种JAR包方式不支持JSP模板。
- 如果使用Undertow嵌入式容器部署Spring Boot项目，也不支持JSP模板。
- Spring Boot默认提供了一个处理请求路径“/error”的统一错误处理器，用于返回具体的异常信息。使用JSP模板时，无法对默认的错误处理器进行覆盖，只能根据Spring Boot要求在指定位置定制错误页面。



4.2

Thymeleaf基础入门



Thymeleaf是Spring Boot官方推荐使用的模板引擎，它创建的模板可维护性非常高，Thymeleaf使开发人员能够在保持设计原型不变的情况下，将程序的数据和逻辑注入到模板文件中，以提高设计团队和开发团队的工作协调性，减少了因理解和实现偏差产生的问题，也使得维护工作更加高效和准确。下面对Thymeleaf简介、Thymeleaf常用属性、标准表达式等基础入门知识进行讲解。

>>> 4.2.1 Thymeleaf简介



黑马程序员
www.itheima.com

传智教育旗下
高端IT教育品牌



了解Thymeleaf简介，能够说出
Thymeleaf是什么和Thymeleaf的特点



Thymeleaf以不影响模板作为设计原型的方式，将其逻辑数据注入到模板文件中，从而在开发团队中实现更强大的协作，支持HTML、XML、TEXT、JAVASCRIPT、CSS、RAW六种模板。

Thymeleaf 模板引擎特点

(1) 动静结合



Thymeleaf可以查看页面的静态效果和动态页面效果。

(2) 开箱即用



Thymeleaf 提供了Spring标准方言，以及与 SpringMVC 完美集成的可选模块。

(4) 与Spring Boot完美整合



Spring Boot为Thymeleaf 整合提供了自动配置支持，并且还Thymeleaf 设置了视图解析器。

(3) 多方言支持



提供了Thymeleaf标准和Spring标准方言，可以直接套用模板实现JSTL、OGNL表达式。

>>> 4.2.2 Thymeleaf常用属性



黑马程序员
www.itheima.com

传智教育旗下
高端IT教育品牌



掌握Thymeleaf常用属性，能够在模板文件中正确使用Thymeleaf常用属性

>>> 4.2.2 Thymeleaf常用属性



黑马程序员
www.itheima.com

传智教育旗下
高端IT教育品牌

Thymeleaf的HTML模板文件中使用的HTML元素语法几乎是标准的HTML语法，Thymeleaf 在标准HTML标签中增加一些格式为 “th: xxx” 的属性以实现 “模板+数据” 的展示。

```
<!DOCTYPE html>
<html lang="en" xmlns:th="http://www.thymeleaf.org">
<head>
  <meta charset="UTF-8">
  <title>Title</title>
</head>
<body>
  <h1 th:text="Hello Thymeleaf">Thymeleaf静态页面</h1>
</body>
</html>
```

名称空间：可避免编辑器出现HTML验证错误

对<h1>标签中的文本进行替换

>>> 4.2.2 Thymeleaf常用属性



黑马程序员
www.itheima.com

传智教育旗下
高端IT教育品牌

当直接使用浏览器打开模板文件。



当启动Web项目后，以访问Web应用中的资源的方式打开文件。





Thymeleaf常用的th属性

属性	作用
th:each	用于遍历，可以遍历集合、数组等对象。
th:if	根据条件判断是否需要展示此标签
th:unless	和 th:if 判断相反，满足条件时不显示
th:switch	与Java的switch语句类似，通常与th:case配合使用，根据不同的条件展示不同的内容
th:case	条件判断，进行选择性匹配
th:object	用于绑定数据对象
th:with	定义局部变量



Thymeleaf常用的th属性

属性	作用
th:action	表单提交地址
th:attr	动态拼接属性值
th:onclick	绑定鼠标单击事件
th:value	替换value属性的值
th:href	用于设定链接地址
th:src	用于替换src属性的值，设定链接地址
th:text	文本替换，转义特殊字符

Thymeleaf常用的th属性

属性	作用
th:utext	文本替换，不转义特殊字符
th:fragment	声明片段，类似 JSP 的 tag，用来定义一段被引用或包含的模板片段
th:insert	用于指定在模板中插入其他模板片段，类似JSP中的include标签，可以通过th:fragment属性定义要被替换进来的片段
th:replace	用于将指定位置的内容替换为指定的模板片段，可以通过th:fragment属性定义要被替换进来的片段
th:block	创建区域块，如果内容为空则不创建

>>> 4.2.3 Thymeleaf标准表达式



黑马程序员
www.itheima.com

传智教育旗下
高端IT教育品牌



掌握Thymeleaf标准表达式，能够在模板文件中正确使用Thymeleaf标准表达式



动态页面与静态页面的主要区别是动态页面中可以动态显示相关的数据，Thymeleaf标准方言中提供了一些可以动态获取数据的标准表达式，这些标准表达式与JSP的EL功能非常相似，可以获取Web Context中的请求参数、请求属性、会话属性、应用属性等数据。

根据语法规则，Thymeleaf标准表达式主要分为变量表达式、选择表达式、消息表达式、链接表达式、分段表达式，下面分别对这几种表达式进行讲解。



4.2.3 Thymeleaf标准表达式



黑马程序员
www.itheima.com

传智教育旗下
高端IT教育品牌

1. 变量表达式

Thymeleaf中使用`${}`包裹内容的表达式被称为变量表达式，使用变量表达式可以访问容器上下文中的变量。

```
<p th:text="${title}">这是标题</p>
```

变量表达式中除了可以直接访问变量，还可以访问变量的属性和方法。

```
<p th:text="${user.name}">这是标题</p>  
<p th:text="${user.info()}">这是标题</p>
```



1. 变量表达式

Thymeleaf为变量所在域提供了一些内置对象。

- #ctx: 上下文对象。
- #vars: 上下文变量。
- #locale: 上下文语言环境。
- #request: HttpServletRequest对象, 仅在Web应用中可用。
- #response: HttpServletResponse对象, 仅在Web应用中可用。
- #session: HttpSession对象, 仅在Web应用中可用。
- #servletContext: ServletContext对象, 仅在Web应用中可用。



4.2.3 Thymeleaf标准表达式



黑马程序员
www.itheima.com

传智教育旗下
高端IT教育品牌

1. 变量表达式

内置对象都以符号#开头，通过变量表达式可以访问这些内置对象，例如，通过变量表达式访问HttpSession中的name属性。

```
<p th:text="${#session.getAttribute('name')}"></p>
```

在变量表达式中也使用内置对象直接访问内置对象的属性。

```
<p th:text="${session.name}"></p>
```



4.2.3 Thymeleaf标准表达式



黑马程序员
www.itheima.com

传智教育旗下
高端IT教育品牌

1. 变量表达式

为了便于开发，Thymeleaf中还提供了一些**内置的工具对象**，这些工具对象就像是Java内置对象一样，可以直接访问对应**Java内置对象的方法**来进行各种操作。

- **#strings**: 字符串工具对象，常用方法有equals()、length()、substring()、replace()等。
- **#numbers**: 数字工具对象，常用的方法有formatDecimal()等。
- **#bools**: 布尔工具对象，常用的方法有isTrue()和isFalse ()等。
- **#arrays**: 数组工具对象，常用的方法有toArray()、isEmpty()、contains()等。
- **#lists**: List集合工具对象，常用的方法有toList()、size()、isEmpty()等。
- **#sets**: Set集合工具对象，常用的方法有toSet()、size()、isEmpty()、contains()等。
- **#maps**: Map 集合工具对象，常用的方法有size()、isEmpty()、containsKey()、containsValue()等。
- **#dates**: 日期工具对象，常用的方法有format()、year()、month()、hour()等。



4.2.3 Thymeleaf标准表达式



黑马程序员
www.itheima.com

传智教育旗下
高端IT教育品牌

1. 变量表达式

通过变量表达式可以使用工具对象，例如，在变量表达式中，使用#strings调用length()方法。

```
<h1 th:text="${#strings.length('thymeleaf')}"></h1>
```



4.2.3 Thymeleaf标准表达式



黑马程序员
www.itheima.com

传智教育旗下
高端IT教育品牌

2.选择表达式

Thymeleaf中使用*{}包裹内容的表达式被称为选择表达式，选择表达式与变量表达式功能基本一致，只是选择表达式计算的是绑定的对象，而不是整个环境变量映射。标签中通过“th:object”属性绑定对象后，可以在该标签的后代标签中使用选择表达式（*{...}）访问该对象中的属性，其中，“*”即代表绑定的对象。

```
<div th:object="${user}" >  
    <p th:text="*{username}">name</p>  
</div>
```

访问user对象中的username属性。



4.2.3 Thymeleaf标准表达式



黑马程序员
www.itheima.com

传智教育旗下
高端IT教育品牌

3.消息表达式

Thymeleaf中使用#{ }包裹内容的表达式被称为消息表达式，消息表达式可以显示静态文本的内容，通常与th:text属性一起使用。消息表达式显示的内容通常是读取配置文件中信息，在#{ }中指定配置文件中的Key，则会在页面中显示配置文件中Key对应的Value。

```
<p th:text="#{location}"></p>
```

读取项目Properties配置文件中location属性的值。

读取指定配置文件的内容时，需要先在项目的全局配置文件中通过spring.messages.basename属性进行文件的指定。



4.链接表达式

Thymeleaf中使用@{}包裹内容的表达式被称为**链接表达式**，表达式中包裹的内容只能写一个绝对/相对URL地址，如果绝对/相对URL地址中包含动态参数，就需要结合**变量表达式**进行**拼接**。

链接表达式中的URL地址的写法

- 以http协议开头的绝对地址。
- 以/开头的相对地址，Thymeleaf会将开头的斜杠(/)解析为当前工程的上下文路径ContextPath，而浏览器会自动为其添加“http://主机名:端口号”。
- 不以/开头的相对地址。

```
<link rel="stylesheet" th:href="@{/css/bootstrap.css}">
```



5.分段表达式

Thymeleaf中使用~{}包裹内容的表达式被称为分段表达式，分段表达式用于在模板页面中引用其他的模板片段，该表达式支持以下两种语法结构。

- ① ~{**templatename**:**fragmentname**} → 片段名, Thymeleaf 通过 th:fragment 声明定义片段
 - ② ~{templatename:**#id**} → id 选择器
- 模版名

```
<div th:insert="~{thymeleafDemo::title}"></div>
```

上述代码中，使用th:insert属性将title片段模板插入到该<div>标签中。thymeleafDemo为模板名称，Thymeleaf会自动查找“/resources/templates/”目录下thymeleafDemo模板中声明的名称为title的片段。

4.3

案例：图书管理

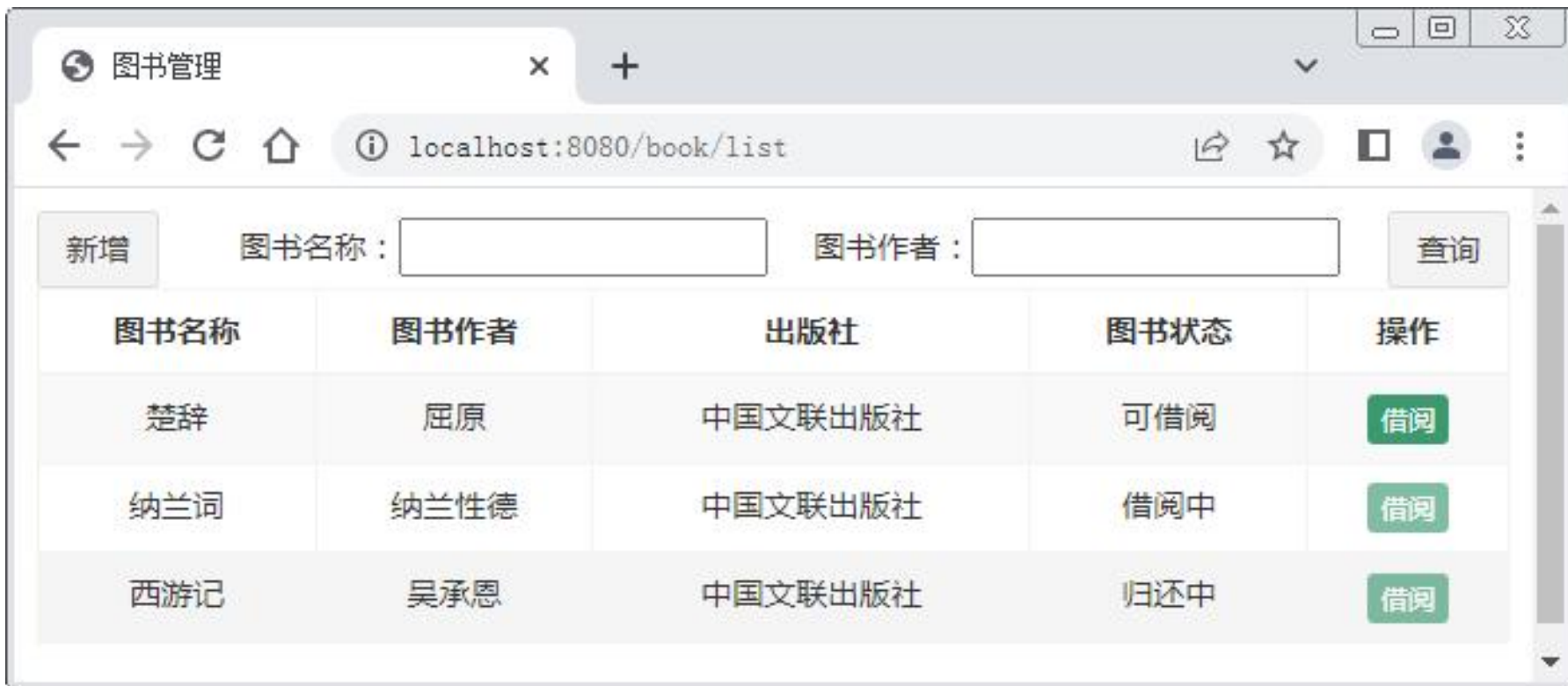




掌握图书管理案例的实现，能够使用
Spring Boot整合Thymeleaf完成图书
管理案例



为了加深大家对Thymeleaf相关知识的理解，下面通过一个图书管理案例演示Spring Boot整合Thymeleaf的应用。





4.3 案例：图书管理



黑马程序员
www.itheima.com

传智教育旗下
高端IT教育品牌

1.任务需求

- (1) 项目使用Spring Boot整合Thymeleaf，项目展示的页面效果全部通过Thymeleaf的模板文件实现。
- (2) 查询所有图书。访问http://localhost:8080/book/list时，查询所有图书，并展示在页面中。
- (3) 选择性显示按钮。当Session中存在用户角色为“ADMIN”时，显示“新增”按钮，否则不显示该按钮。
- (4) 按条件查询图书。单击“查询”按钮时，根据搜索框中的查询条件查询对应的图书信息。
- (5) 借阅图书。当图书状态为可借阅时，对应的“借阅”按钮为可用状态，并且单击“借阅”按钮时，将当前申请借阅图书的编号异步发送到后台。



2.任务实现

(1) Spring Boot整合Thymeleaf

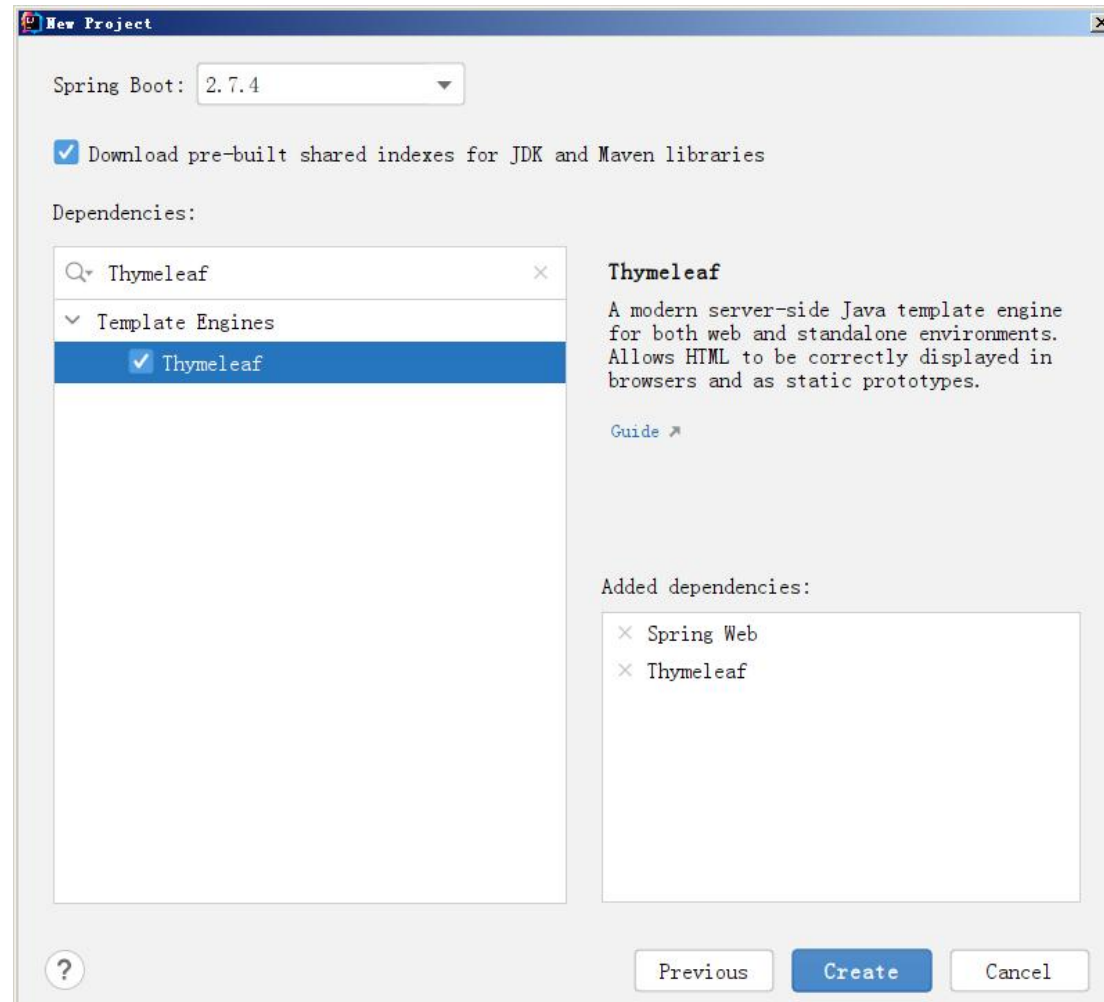
SpringBoot为Thymeleaf 提供了一系列默认配置，Spring Boot项目中只要引入Thymeleaf 的启动器，项目启动时就会自动进行对应配置，因此Spring Boot整合Thymeleaf只需在创建的Spring Boot项目中添加Thymeleaf 启动器即可。



2.任务实现

根据创建Spring Boot项目的方式，在Spring Boot项目中添加Thymeleaf 启动器有2种方法。

①使用Spring Initializr方式构建Spring Boot项目时，在Spring Boot场景依赖选择界面添加Thymeleaf依赖。





4.3 案例：图书管理



黑马程序员
www.itheima.com

传智教育旗下
高端IT教育品牌

2.任务实现

②使用Maven方式构建Spring Boot项目后，在pom.xml文件中引入Thymeleaf 启动器。同时项目需要使用Spring MVC。

```
<dependencies>
  <dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-thymeleaf</artifactId>
  </dependency>
  <dependency>
    <groupId>org.springframework.boot</groupId>
    <artifactId>spring-boot-starter-web</artifactId>
  </dependency>
</dependencies>
```



2.任务实现

(2) 创建实体类

图书管理页面中包含2个实体对象，分别为用户实体和图书实体，并且当前还没有学习Spring Boot和持久层技术的整合，在此，选择通过一个类封装一些图书数据，页面展示的图书信息都从这个类中获取数据。需要创建用户类、图书类和数据类，具体如文件4-1~文件4-3所示。



源代码

文件4-1 [User.java](#)

文件4-2 [Book.java](#)

文件4-3 [Data.java](#)





2.任务实现

(3) 创建控制器类

创建控制器类，在控制器类中创建对应的方法接收和响应页面发送的请求，具体代码如文件4-4所示。



源代码

文件4-4

[BookController.java](#)





2.任务实现

(4) 创建模板文件

在resources\templates目录下创建名称为books的HTML文件作为模板文件，并将模板文件中通过Thymeleaf的属性和标准表达式引入样式文件，以及获取数据进行展示，具体代码如文件4-5所示。



源 代 码

文件4-5
[books.html](#)





2.任务实现

(5) 测试图书管理

启动项目，在浏览器中访问<http://localhost:8080/book/list>查询所有图书信息。

图书名称	图书作者	出版社	图书状态	操作
楚辞	屈原	中国文联出版社	可借阅	借阅
纳兰词	纳兰性德	中国文联出版社	借阅中	借阅
西游记	吴承恩	中国文联出版社	归还中	借阅



2.任务实现

(5) 测试图书管理

在搜索框中输入查询条件后单击“查询”按钮，按条件查询图书信息，以查询图书名称中包含“辞”字的图书信息为例。

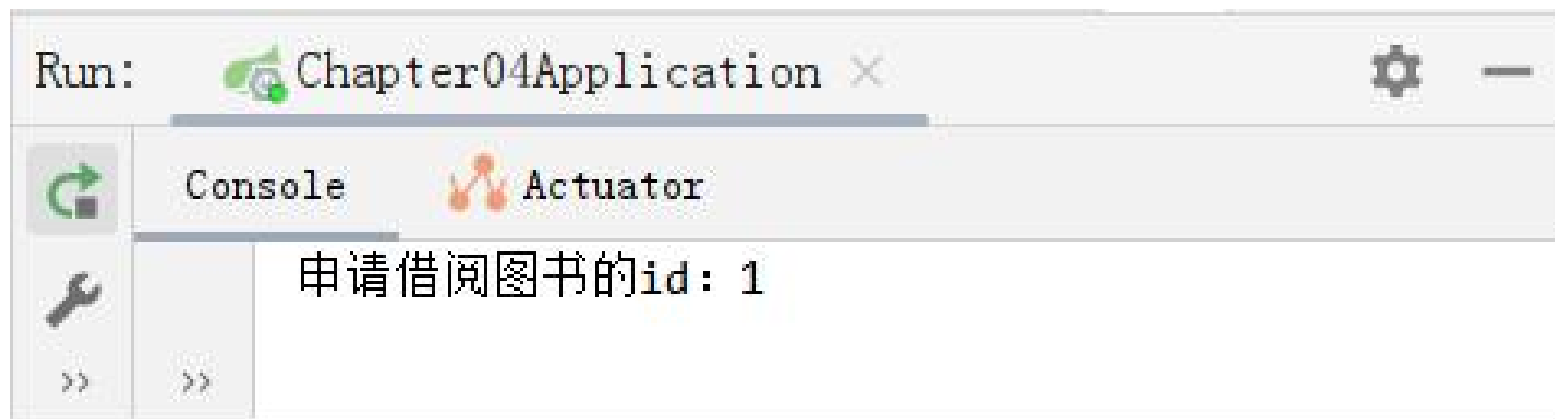
图书名称	图书作者	出版社	图书状态	操作
楚辞	屈原	中国文联出版社	可借阅	借阅



2.任务实现

(5) 测试图书管理

单击图书《楚辞》的“借阅”按钮，此时IDEA控制台中输出图书对应的id。





本章小结

本章主要对Spring Boot整合Thymeleaf进行了讲解。首先讲解了Spring Boot支持的模板引擎；然后讲解了Thymeleaf基础入门；最后通过图书管理案例演示Spring Boot整合Thymeleaf，以及Thymeleaf常用属性和标准表达式的应用。通过本章的学习，希望大家可以对Spring Boot整合Thymeleaf有所了解，为后续更深入学习Spring Boot做好铺垫。

為千萬學生少走彎路而著書



黑马程序员
www.itheima.com

传智教育旗下
高端IT教育品牌