

FreeMarker学习笔记

FreeMarker

介绍

性能特点

快速入门

创建spring boot工程

添加依赖

配置文件

创建实体类

创建模板

创建controller

运行测试

FreeMarker 基础指令

List指令

创建test2.ftl

修改controller

运行测试

遍历Map数据

创建test3.ftl

修改controller

运行测试

if指令

创建test4.ftl

修改controller

运行测试

运算符

空值处理

创建test5.ftl

修改controller

运行测试

内置对象

配置

创建test6.ftl

修改controller

运行测试

FreeMarker

介绍

FreeMarker是一款模板引擎：即一种基于模板和要改变的数据，并用来生成输出文本（HTML网页、电子邮件、配置文件、源代码等）的通用工具。它不是面向最终用户的，而是一个Java类库，是一款程序员可以嵌入他们所开发产品的组件。

FreeMarker是免费的，基于Apache许可证2.0版本发布。其模板编写为FreeMarker Template Language (FTL)，属于简单、专用的语言。需要准备数据在真实编程语言中来显示，比如数据库查询和业务运算，之后模板显示已经准备好的数据。在模板中，主要用于如何展现数据，而在模板之外注意于要展示什么数据

常用的java模板引擎还有哪些？

Jsp、Freemarker、Thymeleaf、Velocity 等

性能特点

模板并没有包含程序逻辑来查找当前的访问者是谁，或者去查询数据库获取最新的产品。显示的数据是在FreeMarker之外准备的，通常是一些“真正的”编程语言（比如Java）所编写的代码。模板作者无需知道这些值是如何计算出的。事实上，这些值的计算方式可以完全被修改，而模板可以保持不变，而且页面的样式也可以完全被修改而无需改动模板。当模板作者（设计师）和程序员不是同一人时，显示逻辑和业务逻辑相分离的做法是非常有用的，即便模板作者和程序员是一个人，这么来做也会帮助管理应用程序的复杂性。保证模板专注于显示问题（视觉设计，布局和格式化）是高效使用模板引擎的关键

1. 通用性

能够生成各种文本：HTML、XML、RTF、Java源代码等等。

易于嵌入到产品中：轻量级；不需要Servlet环境。

插件式模板载入器：可以从任何源载入模板，如本地文件、数据库等等。

可以按所需生成文本：保存到本地文件；作为Email发送；从Web应用程序发送它返回给Web浏览器。

2. 模板语言

所有常用的指令：include、if/elseif/else、循环结构。

在模板中创建和改变变量。

几乎在任何地方都可以使用复杂表达式来指定值。

命名的宏，可以具有位置参数和嵌套内容。

名字空间有助于建立和维护可重用的宏库，或者将一个大工程分成模块，而不必担心名字冲突。

输出转换块：在嵌套模板片段生成输出时，转换HTML转义、压缩、语法高亮等等；可以定义自己的转换。

3. 通用数据模型

FreeMarker不是直接反射到Java对象，Java对象通过插件式对象封装，以变量方式在模板中显示。

可以使用抽象（接口）方式表示对象（JavaBean、XML文档、SQL查询结果集等等），告诉模板开发者使用。方法，使其不受技术细节的打扰。

xml配置

xml配置

4. 为Web准备

在模板语言中内建处理典型Web相关任务（如HTML转义）的结构。

能够集成到Model2 Web应用框架中作为JSP的替代。

支持JSP标记库。

为MVC模式设计：分离可视化设计和应用程序逻辑；分离页面设计员和程序员。

5. 智能的国际化和本地化

字符集智能化（内部使用UNICODE）。

数字格式本地化敏感。

日期和时间格式本地化敏感。

非US字符集可以用作标识（如变量名）。

多种不同语言的相同模板。

6. XML处理能力

<#recurse> 和<#visit>指令（2.3版本）用于递归遍历XML树。

在模板中清楚和直接的访问XML对象模型。

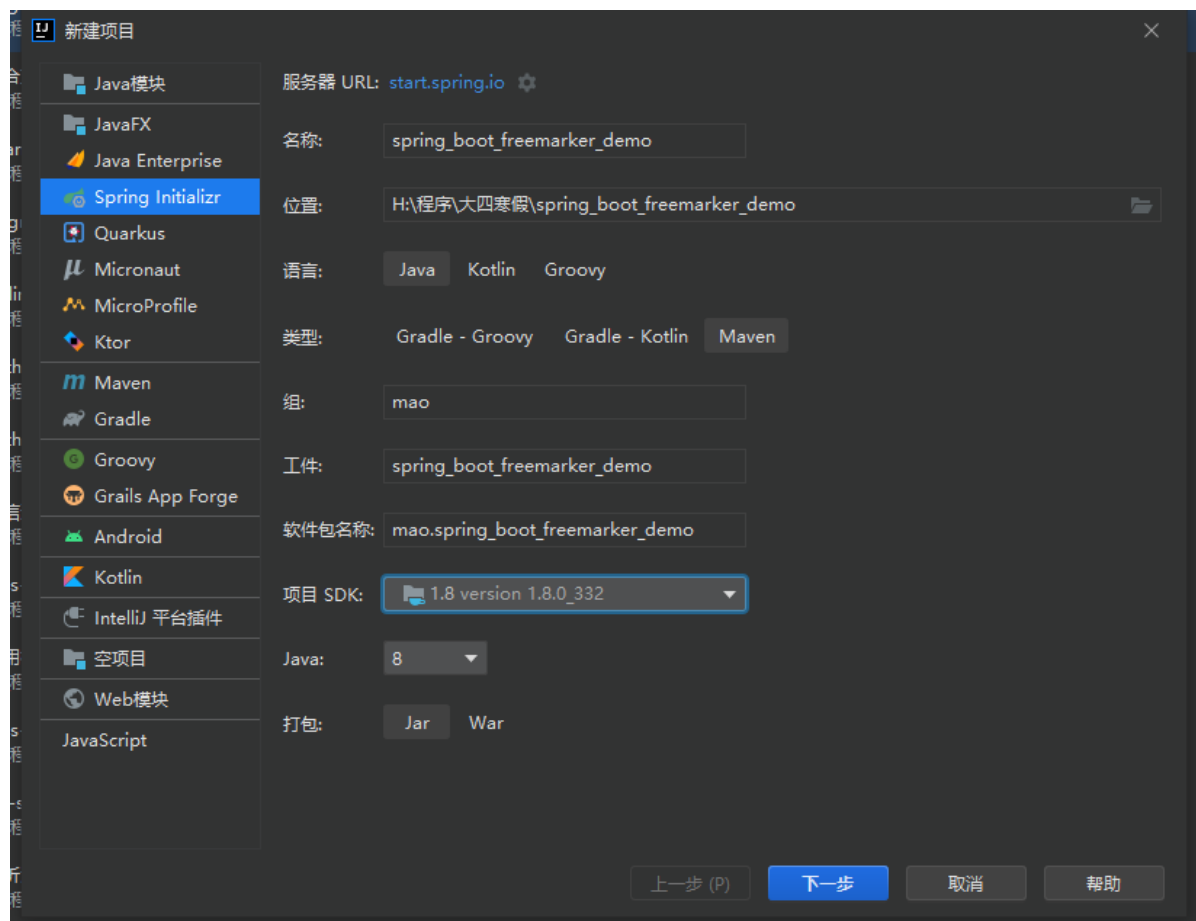
快速入门

freemarker作为springmvc一种视图格式，默认情况下SpringMVC支持freemarker视图格式

需要创建Spring Boot+Freemarker工程用于测试模板。

创建spring boot工程

项目名称为spring_boot_freemarker_demo



添加依赖

```
1 <!--freemarker 模板引擎-->
2 <dependency>
3     <groupId>org.springframework.boot</groupId>
4     <artifactId>spring-boot-starter-freemarker</artifactId>
5     <version>3.0.2</version>
6 </dependency>
```

```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <project xmlns="http://maven.apache.org/POM/4.0.0"
3     xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
4     xsi:schemaLocation="http://maven.apache.org/POM/4.0.0
5     http://maven.apache.org/xsd/maven-4.0.0.xsd">
6     <modelVersion>4.0.0</modelVersion>
7     <parent>
8         <groupId>org.springframework.boot</groupId>
9         <artifactId>spring-boot-starter-parent</artifactId>
10        <version>2.7.1</version>
11        <relativePath/> <!-- lookup parent from repository -->
12    </parent>
```

```

13     <groupId>mao</groupId>
14     <artifactId>spring_boot_freemarker_demo</artifactId>
15     <version>0.0.1-SNAPSHOT</version>
16     <name>spring_boot_freemarker_demo</name>
17     <description>spring_boot_freemarker_demo</description>
18
19     <properties>
20         <java.version>1.8</java.version>
21     </properties>
22
23     <dependencies>
24
25         <dependency>
26             <groupId>org.springframework.boot</groupId>
27             <artifactId>spring-boot-starter-web</artifactId>
28         </dependency>
29
30         <dependency>
31             <groupId>org.springframework.boot</groupId>
32             <artifactId>spring-boot-starter-test</artifactId>
33             <scope>test</scope>
34         </dependency>
35
36         <!--freemarker 模板引擎-->
37         <dependency>
38             <groupId>org.springframework.boot</groupId>
39             <artifactId>spring-boot-starter-freemarker</artifactId>
40             <version>3.0.2</version>
41         </dependency>
42
43     </dependencies>
44
45     <build>
46         <plugins>
47             <plugin>
48                 <groupId>org.springframework.boot</groupId>
49                 <artifactId>spring-boot-maven-plugin</artifactId>
50             </plugin>
51         </plugins>
52     </build>
53
54 </project>

```

配置文件

```

1  spring:
2    freemarker:
3      # 后缀名
4      suffix: .ftl
5      content-type: text/html
6      enabled: true

```

```

7      # 缓存配置
8      cache: true
9      # 模板加载路径 按需配置（这里就是我们上面的忽略配置）
10     template-loader-path: classpath:/templates/
11     # 编码格式
12     charset: UTF-8
13     settings:
14         # 数字格式化，无小数点
15         number_format: '0.##'

```

创建实体类

在freemarker的测试工程下创建模型类型用于测试

```

1  package mao.spring_boot_freemarker_demo.entity;
2
3  /**
4   * Project name(项目名称): spring_boot_freemarker_demo
5   * Package(包名): mao.spring_boot_freemarker_demo.entity
6   * Class(类名): Student
7   * Author(作者): mao
8   * Author QQ: 1296193245
9   * Github: https://github.com/maomao124/
10  * Date(创建日期): 2023/1/29
11  * Time(创建时间): 19:47
12  * Version(版本): 1.0
13  * Description(描述): 无
14  */
15
16  public class Student
17  {
18      /**
19       * id
20       */
21      private Long id;
22
23      /**
24       * 名字
25       */
26      private String name;
27
28      /**
29       * 性别
30       */
31      private String sex;
32
33      /**
34       * 年龄
35       */
36      private int age;

```

```

37
38
39     /**
40      * 学生
41      */
42     public Student()
43     {
44
45     }
46
47     /**
48      * 学生
49      *
50      * @param id    id
51      * @param name  名字
52      * @param sex   性别
53      * @param age   年龄
54      */
55     public Student(Long id, String name, String sex, int age)
56     {
57         this.id = id;
58         this.name = name;
59         this.sex = sex;
60         this.age = age;
61     }
62
63     /**
64      * 得到id
65      *
66      * @return {@link Long}
67      */
68     public Long getId()
69     {
70         return id;
71     }
72
73     /**
74      * 设置id
75      *
76      * @param id id
77      */
78     public void setId(Long id)
79     {
80         this.id = id;
81     }
82
83     /**
84      * 得到名字
85      *
86      * @return {@link String}
87      */
88     public String getName()
89     {
90         return name;
91     }
92
93     /**
94      * 设置名字

```

```

95     *
96     * @param name 名字
97     */
98     public void setName(String name)
99     {
100         this.name = name;
101     }
102
103     /**
104     * 得到性别
105     *
106     * @return {@link String}
107     */
108     public String getSex()
109     {
110         return sex;
111     }
112
113     /**
114     * 设置性别
115     *
116     * @param sex 性别
117     */
118     public void setSex(String sex)
119     {
120         this.sex = sex;
121     }
122
123     /**
124     * 得到年龄
125     *
126     * @return int
127     */
128     public int getAge()
129     {
130         return age;
131     }
132
133     /**
134     * 设置年龄
135     *
136     * @param age 年龄
137     */
138     public void setAge(int age)
139     {
140         this.age = age;
141     }
142
143     @Override
144     @SuppressWarnings("all")
145     public String toString()
146     {
147         final StringBuilder stringbuilder = new StringBuilder();
148         stringbuilder.append("id: ").append(id).append('\n');
149         stringbuilder.append("name: ").append(name).append('\n');
150         stringbuilder.append("sex: ").append(sex).append('\n');
151         stringbuilder.append("age: ").append(age).append('\n');
152         return stringbuilder.toString();

```



```
153     }
154 }
```

创建模板

在 src/main/resources 下创建 templates，此目录为 freemarker 的默认模板存放目录

在 templates 下创建模板文件 test1.ftl，模板中的 `${name}` 最终会被 freemarker 替换成具体的数据

```
1 <!DOCTYPE html>
2
3 <html lang="en">
4 <head>
5     <meta charset="UTF-8">
6     <title>Title</title>
7 </head>
8 <body>
9
10 <h1>hello! my name is ${name}</h1>
11
12 </body>
13 </html>
```

创建controller

创建 Controller 类，向 Map 中添加 name，最后返回模板文件

```
1 package mao.spring_boot_freemarker_demo;
2
3 import org.springframework.stereotype.Controller;
4 import org.springframework.web.bind.annotation.RequestMapping;
5
6 import javax.servlet.http.HttpServletRequest;
7 import java.util.Map;
8
9 /**
10  * Project name(项目名称): spring_boot_freemarker_demo
11  * Package(包名): mao.spring_boot_freemarker_demo
12  * Class(类名): TestController
13  * Author(作者): mao
14  * Author QQ: 1296193245
15  * GitHub: https://github.com/maomao124/
16  * Date(创建日期): 2023/1/29
17  * Time(创建时间): 19:56
```

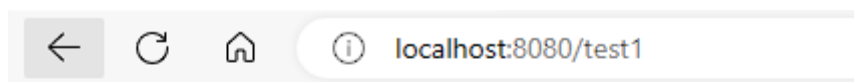
```

18  * Version(版本): 1.0
19  * Description(描述): 无
20  */
21
22  @Controller
23  public class TestController
24  {
25      @RequestMapping("/test1")
26      public String freemarker(Map<String, Object> map)
27      {
28          map.put("name", "张三");
29          //返回模板文件名称
30          return "test1";
31      }
32  }

```

运行测试

<http://localhost:8080/test1>



hello! my name is 张三

FreeMarker 基础指令

- 注释，即<#--和-->，介于其之间的内容会被freemarker忽略
- 插值 (Interpolation)：即\${..}部分，freemarker会用真实的值代替\${..}
- FTL指令：和HTML标记类似，名字前加#予以区分，Freemarker会解析标签中的表达式或逻辑
- 文本，仅文本信息，这些不是freemarker的注释、插值、FTL指令的内容会被freemarker忽略解析，直接输出内容

List指令

- `_index`：得到循环的下标，使用方法是在`stu`后边加"`_index`"，它的值是从0开始

创建test2.ftl

```
1 <!DOCTYPE html>
2
3 <html lang="en">
4 <head>
5     <meta charset="UTF-8">
6     <title>Title</title>
7 </head>
8 <body>
9
10 <table>
11     <tr>
12         <td>序号</td>
13         <td>学号</td>
14         <td>姓名</td>
15         <td>性别</td>
16         <td>年龄</td>
17     </tr>
18     <#list stus as stu>
19         <tr>
20             <td>${stu_index + 1}</td>
21             <td>${stu.id}</td>
22             <td>${stu.name}</td>
23             <td>${stu.sex}</td>
24             <td>${stu.age}</td>
25         </tr>
26     </#list>
27 </table>
28
29 </body>
30 </html>
```

修改controller

```
1 package mao.spring_boot_freemarker_demo;
2
3 import mao.spring_boot_freemarker_demo.entity.Student;
4 import org.springframework.stereotype.Controller;
5 import org.springframework.web.bind.annotation.RequestMapping;
6
7 import javax.servlet.http.HttpServletRequest;
8 import java.util.ArrayList;
9 import java.util.List;
10 import java.util.Map;
11
12 /**
13  * Project name(项目名称): spring_boot_freemarker_demo
14  * Package(包名): mao.spring_boot_freemarker_demo
15  * Class(类名): TestController
16  * Author(作者): mao
17  * Author QQ: 1296193245
18  * GitHub: https://github.com/maomao124/
```

```

19  * Date(创建日期): 2023/1/29
20  * Time(创建时间): 19:56
21  * Version(版本): 1.0
22  * Description(描述): 无
23  */
24
25  @Controller
26  public class TestController
27  {
28      @RequestMapping("/test1")
29      public String freemarker(Map<String, Object> map)
30      {
31          map.put("name", "张三");
32          //返回模板文件名称
33          return "test1";
34      }
35
36      @RequestMapping("/test2")
37      public String freemarker2(Map<String, Object> map)
38      {
39          Student student1 = new Student();
40          student1.setId(10001L);
41          student1.setName("张三");
42          student1.setSex("男");
43          student1.setAge(18);
44
45          Student student2 = new Student();
46          student2.setId(10002L);
47          student2.setName("李四");
48          student2.setSex("男");
49          student2.setAge(19);
50
51          Student student3 = new Student();
52          student3.setId(10004L);
53          student3.setName("王五");
54          student3.setSex("女");
55          student3.setAge(17);
56
57          List<Student> list = new ArrayList<>(3);
58          list.add(student1);
59          list.add(student2);
60          list.add(student3);
61
62          map.put("stus", list);
63
64          //返回模板文件名称
65          return "test2";
66      }
67  }

```

运行测试

<http://localhost:8080/test2>

序号	学号	姓名	性别	年龄
1	10001	张三	男	18
2	10002	李四	男	19
3	10004	王五	女	17

遍历Map数据

创建test3.ftl

```
1 <!DOCTYPE html>
2
3 <html lang="en">
4 <head>
5     <meta charset="UTF-8">
6     <title>Title</title>
7 </head>
8 <body>
9
10 <table>
11     <tr>
12         <td>序号</td>
13         <td>学号</td>
14         <td>姓名</td>
15         <td>性别</td>
16         <td>年龄</td>
17     </tr>
18     <#list studentMap?keys as k>
19         <tr>
20             <td>${k_index + 1}</td>
21             <td>${studentMap[k].id}</td>
22             <td>${studentMap[k].name}</td>
23             <td>${studentMap[k].sex}</td>
24             <td>${studentMap[k].age}</td>
25         </tr>
26     </#list>
27 </table>
28
29 <br>
30
```

```
31  ${studentMap['student1'].name}
32  <br>
33  ${studentMap.student2.name}
34
35  </body>
36  </html>
```

修改controller

```
1  package mao.spring_boot_freemarker_demo;
2
3  import mao.spring_boot_freemarker_demo.entity.Student;
4  import org.springframework.stereotype.Controller;
5  import org.springframework.web.bind.annotation.RequestMapping;
6
7  import javax.servlet.http.HttpServletRequest;
8  import java.util.ArrayList;
9  import java.util.HashMap;
10 import java.util.List;
11 import java.util.Map;
12
13 /**
14  * Project name(项目名称): spring_boot_freemarker_demo
15  * Package(包名): mao.spring_boot_freemarker_demo
16  * Class(类名): TestController
17  * Author(作者): mao
18  * Author QQ: 1296193245
19  * GitHub: https://github.com/maomao124/
20  * Date(创建日期): 2023/1/29
21  * Time(创建时间): 19:56
22  * Version(版本): 1.0
23  * Description(描述): 无
24  */
25
26 @Controller
27 public class TestController
28 {
29     @RequestMapping("/test1")
30     public String freemarker(Map<String, Object> map)
31     {
32         map.put("name", "张三");
33         //返回模板文件名称
34         return "test1";
35     }
36
37     @RequestMapping("/test2")
38     public String freemarker2(Map<String, Object> map)
39     {
40         Student student1 = new Student();
41         student1.setId(10001L);
42         student1.setName("张三");
43         student1.setSex("男");
44         student1.setAge(18);
```

```

45
46     Student student2 = new Student();
47     student2.setId(10002L);
48     student2.setName("李四");
49     student2.setSex("男");
50     student2.setAge(19);
51
52     Student student3 = new Student();
53     student3.setId(10004L);
54     student3.setName("王五");
55     student3.setSex("女");
56     student3.setAge(17);
57
58     List<Student> list = new ArrayList<>(3);
59     list.add(student1);
60     list.add(student2);
61     list.add(student3);
62
63     map.put("stus", list);
64
65     //返回模板文件名称
66     return "test2";
67 }
68
69 @RequestMapping("/test3")
70 public String freemarker3(Map<String, Object> map)
71 {
72     Student student1 = new Student();
73     student1.setId(10001L);
74     student1.setName("张三");
75     student1.setSex("男");
76     student1.setAge(18);
77
78     Student student2 = new Student();
79     student2.setId(10002L);
80     student2.setName("李四");
81     student2.setSex("男");
82     student2.setAge(19);
83
84     Student student3 = new Student();
85     student3.setId(10004L);
86     student3.setName("王五");
87     student3.setSex("女");
88     student3.setAge(17);
89
90     Map<String, Student> studentMap = new HashMap<>();
91
92     studentMap.put("student1", student1);
93     studentMap.put("student2", student2);
94     studentMap.put("student3", student3);
95
96     map.put("studentMap", studentMap);
97
98     //返回模板文件名称
99     return "test3";
100 }
101 }

```

运行测试

<http://localhost:8080/test3>

序号	学号	姓名	性别	年龄
1	10004	王五	女	17
2	10002	李四	男	19
3	10001	张三	男	18

张三
李四

if指令

if 指令即判断指令，是常用的FTL指令，freemarker在解析时遇到if会进行判断，条件为真则输出if中间的内容，否则跳过内容不再输出

创建test4.ftl

```
1 <!DOCTYPE html>
2
3 <html lang="en">
4 <head>
5     <meta charset="UTF-8">
6     <title>Title</title>
7 </head>
8 <body>
9
10 <table>
11     <tr>
12         <td>序号</td>
13         <td>学号</td>
14         <td>姓名</td>
15         <td>性别</td>
16         <td>年龄</td>
17     </tr>
18     <#list stus as stu>
19         <tr>
```



```

20         <td <#if stu_index==2>style="color: hotpink" </#if>>${stu_index
+ 1}</td>
21         <td>${stu.id}</td>
22         <td<#if stu.name=='张三'> style="color: skyblue"
</#if>>${stu.name}</td>
23         <td>${stu.sex}</td>
24         <td>${stu.age}</td>
25     </tr>
26 </#list>
27 </table>
28
29 </body>
30 </html>

```

修改controller

```

1  package mao.spring_boot_freemarker_demo;
2
3  import mao.spring_boot_freemarker_demo.entity.Student;
4  import org.springframework.stereotype.Controller;
5  import org.springframework.web.bind.annotation.RequestMapping;
6
7  import javax.servlet.http.HttpServletRequest;
8  import java.util.ArrayList;
9  import java.util.HashMap;
10 import java.util.List;
11 import java.util.Map;
12
13 /**
14  * Project name(项目名称): spring_boot_freemarker_demo
15  * Package(包名): mao.spring_boot_freemarker_demo
16  * Class(类名): TestController
17  * Author(作者): mao
18  * Author QQ: 1296193245
19  * GitHub: https://github.com/maomao124/
20  * Date(创建日期): 2023/1/29
21  * Time(创建时间): 19:56
22  * Version(版本): 1.0
23  * Description(描述): 无
24  */
25
26 @Controller
27 public class TestController
28 {
29     @RequestMapping("/test1")
30     public String freemarker(Map<String, Object> map)
31     {
32         map.put("name", "张三");
33         //返回模板文件名称
34         return "test1";
35     }
36
37     @RequestMapping("/test2")

```

```

38     public String freemarker2(Map<String, Object> map)
39     {
40         Student student1 = new Student();
41         student1.setId(10001L);
42         student1.setName("张三");
43         student1.setSex("男");
44         student1.setAge(18);
45
46         Student student2 = new Student();
47         student2.setId(10002L);
48         student2.setName("李四");
49         student2.setSex("男");
50         student2.setAge(19);
51
52         Student student3 = new Student();
53         student3.setId(10004L);
54         student3.setName("王五");
55         student3.setSex("女");
56         student3.setAge(17);
57
58         List<Student> list = new ArrayList<>(3);
59         list.add(student1);
60         list.add(student2);
61         list.add(student3);
62
63         map.put("stus", list);
64
65         //返回模板文件名称
66         return "test2";
67     }
68
69     @RequestMapping("/test3")
70     public String freemarker3(Map<String, Object> map)
71     {
72         Student student1 = new Student();
73         student1.setId(10001L);
74         student1.setName("张三");
75         student1.setSex("男");
76         student1.setAge(18);
77
78         Student student2 = new Student();
79         student2.setId(10002L);
80         student2.setName("李四");
81         student2.setSex("男");
82         student2.setAge(19);
83
84         Student student3 = new Student();
85         student3.setId(10004L);
86         student3.setName("王五");
87         student3.setSex("女");
88         student3.setAge(17);
89
90         Map<String, Student> studentMap = new HashMap<>();
91
92         studentMap.put("student1", student1);
93         studentMap.put("student2", student2);
94         studentMap.put("student3", student3);
95

```

```

96         map.put("studentMap", studentMap);
97
98         //返回模板文件名称
99         return "test3";
100     }
101
102     @RequestMapping("/test4")
103     public String freemarker4(Map<String, Object> map)
104     {
105         Student student1 = new Student();
106         student1.setId(10001L);
107         student1.setName("张三");
108         student1.setSex("男");
109         student1.setAge(18);
110
111         Student student2 = new Student();
112         student2.setId(10002L);
113         student2.setName("李四");
114         student2.setSex("男");
115         student2.setAge(19);
116
117         Student student3 = new Student();
118         student3.setId(10004L);
119         student3.setName("王五");
120         student3.setSex("女");
121         student3.setAge(17);
122
123         List<Student> list = new ArrayList<>(3);
124         list.add(student1);
125         list.add(student2);
126         list.add(student3);
127
128         map.put("stus", list);
129
130         //返回模板文件名称
131         return "test4";
132     }
133 }

```

运行测试

<http://localhost:8080/test4>

序号	学号	姓名	性别	年龄
1	10001	张三	男	18
2	10002	李四	男	19
3	10004	王五	女	17

运算符

- 算数运算符 FreeMarker表达式中完全支持算术运算,FreeMarker支持的算术运算符包括:+, -, *, /, %
- 逻辑运算符 逻辑运算符有如下几个: 逻辑与:&& 逻辑或:|| 逻辑非:! 逻辑运算符只能作用于布尔值,否则将产生错误
- 比较运算符 表达式中支持的比较运算符有如下几个:
 - =或者==:判断两个值是否相等
 - !=:判断两个值是否不等
 - > 或者gt:判断左边值是否大于右边值
 - >=或者gte:判断左边值是否大于等于右边值
 - <或者lt:判断左边值是否小于右 边值
 - <=或者lte:判断左边值是否小于等于右边值

注意: =和!=可以用于字符串,数值和日期来比较是否相等,但=和!=两边必须是相同类型的值,否则会产生错误,而且 FreeMarker是精确比较,"x","x ","X"是不等的.其它的运行符可以作用于数字和日期,但不能作用于字符串,大部分的时候,使用gt等字母运算符代替>会有更好的效果,因为 FreeMarker会把>解释成FTL标签的结束字符,当然,也可以使用括 号来避免这种情况,如:<#if (x>y)>

空值处理

判断某变量是否存在使用 "??" 用法为:variable??.如果该变量存在,返回true,否则返回false

缺失变量默认值使用 "!" 使用!要以指定一个默认值, 当变量为空时显示默认值

如果是嵌套对象则建议使用 () 括起来

为防止stus为空报错可以加上if判断

创建test5.ftl

```
1  <!DOCTYPE html>
2
3  <html lang="en">
4  <head>
5      <meta charset="UTF-8">
6      <title>Title</title>
7  </head>
8  <body>
9
10 <#if stus??>
11
12     <table>
13         <tr>
14             <td>序号</td>
15             <td>学号</td>
16             <td>姓名</td>
17             <td>性别</td>
18             <td>年龄</td>
19         </tr>
20         <#list stus as stu>
21             <tr>
22                 <td <#if stu_index==2>style="color: hotpink"
23 </#if>>${stu_index + 1}</td>
24                 <td>${stu.id}</td>
25                 <td>#if stu.name=='张三'> style="color: skyblue"
26 </#if>>${stu.name}</td>
27                 <td>${stu.sex}</td>
28                 <td>${(stu.age)!}</td>
29             </tr>
30         </#list>
31     </table>
32
33 </#if>
34
35 <#if !stus??>
36     变量 stus 为null
37 </#if>
38
39 </body>
40 </html>
```

修改controller

```
1  package mao.spring_boot_freemarker_demo;
2
3  import mao.spring_boot_freemarker_demo.entity.Student;
4  import org.springframework.stereotype.Controller;
5  import org.springframework.web.bind.annotation.RequestMapping;
```

```

6
7 import javax.servlet.http.HttpServletRequest;
8 import java.util.ArrayList;
9 import java.util.HashMap;
10 import java.util.List;
11 import java.util.Map;
12
13 /**
14  * Project name(项目名称): spring_boot_freemarker_demo
15  * Package(包名): mao.spring_boot_freemarker_demo
16  * Class(类名): TestController
17  * Author(作者): mao
18  * Author QQ: 1296193245
19  * GitHub: https://github.com/maomao124/
20  * Date(创建日期): 2023/1/29
21  * Time(创建时间): 19:56
22  * Version(版本): 1.0
23  * Description(描述): 无
24  */
25
26 @Controller
27 public class TestController
28 {
29     @RequestMapping("/test1")
30     public String freemarker(Map<String, Object> map)
31     {
32         map.put("name", "张三");
33         //返回模板文件名称
34         return "test1";
35     }
36
37     @RequestMapping("/test2")
38     public String freemarker2(Map<String, Object> map)
39     {
40         Student student1 = new Student();
41         student1.setId(10001L);
42         student1.setName("张三");
43         student1.setSex("男");
44         student1.setAge(18);
45
46         Student student2 = new Student();
47         student2.setId(10002L);
48         student2.setName("李四");
49         student2.setSex("男");
50         student2.setAge(19);
51
52         Student student3 = new Student();
53         student3.setId(10004L);
54         student3.setName("王五");
55         student3.setSex("女");
56         student3.setAge(17);
57
58         List<Student> list = new ArrayList<>(3);
59         list.add(student1);
60         list.add(student2);
61         list.add(student3);
62
63         map.put("stus", list);

```

```

64
65         //返回模板文件名称
66         return "test2";
67     }
68
69     @RequestMapping("/test3")
70     public String freemarker3(Map<String, Object> map)
71     {
72         Student student1 = new Student();
73         student1.setId(10001L);
74         student1.setName("张三");
75         student1.setSex("男");
76         student1.setAge(18);
77
78         Student student2 = new Student();
79         student2.setId(10002L);
80         student2.setName("李四");
81         student2.setSex("男");
82         student2.setAge(19);
83
84         Student student3 = new Student();
85         student3.setId(10004L);
86         student3.setName("王五");
87         student3.setSex("女");
88         student3.setAge(17);
89
90         Map<String, Student> studentMap = new HashMap<>();
91
92         studentMap.put("student1", student1);
93         studentMap.put("student2", student2);
94         studentMap.put("student3", student3);
95
96         map.put("studentMap", studentMap);
97
98         //返回模板文件名称
99         return "test3";
100     }
101
102     @RequestMapping("/test4")
103     public String freemarker4(Map<String, Object> map)
104     {
105         Student student1 = new Student();
106         student1.setId(10001L);
107         student1.setName("张三");
108         student1.setSex("男");
109         student1.setAge(18);
110
111         Student student2 = new Student();
112         student2.setId(10002L);
113         student2.setName("李四");
114         student2.setSex("男");
115         student2.setAge(19);
116
117         Student student3 = new Student();
118         student3.setId(10004L);
119         student3.setName("王五");
120         student3.setSex("女");
121         student3.setAge(17);

```

```

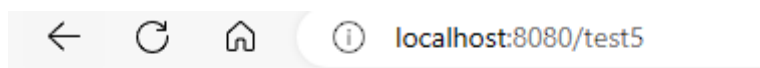
122
123     List<Student> list = new ArrayList<>(3);
124     list.add(student1);
125     list.add(student2);
126     list.add(student3);
127
128     map.put("stus", list);
129
130     //返回模板文件名称
131     return "test4";
132 }
133
134 @RequestMapping("/test5")
135 public String freemarker5(Map<String, Object> map)
136 {
137     Student student1 = new Student();
138     student1.setId(10001L);
139     student1.setName("张三");
140     student1.setSex("男");
141     student1.setAge(18);
142
143     Student student2 = new Student();
144     student2.setId(10002L);
145     student2.setName("李四");
146     student2.setSex("男");
147     //一定概率为空
148     if (Math.random() > 0.5)
149     {
150         student2.setAge(19);
151     }
152     Student student3 = new Student();
153     student3.setId(10004L);
154     student3.setName("王五");
155     student3.setSex("女");
156     //一定概率为空
157     if (Math.random() > 0.5)
158     {
159         student3.setAge(17);
160     }
161
162
163     List<Student> list = new ArrayList<>(3);
164     list.add(student1);
165     list.add(student2);
166     list.add(student3);
167
168     //一定概率为空
169     if (Math.random() > 0.5)
170     {
171         map.put("stus", list);
172     }
173
174     //返回模板文件名称
175     return "test5";
176 }
177 }

```


运行测试

<http://localhost:8080/test5>

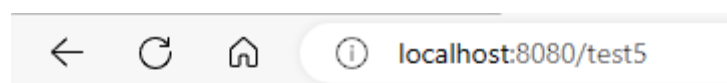
多刷新几次查看结果



变量 stus 为null



序号	学号	姓名	性别	年龄
1	10001	张三	男	18
2	10002	李四	男	0
3	10004	王五	女	0



序号	学号	姓名	性别	年龄
1	10001	张三	男	18
2	10002	李四	男	19
3	10004	王五	女	0



序号	学号	姓名	性别	年龄
1	10001	张三	男	18
2	10002	李四	男	19
3	10004	王五	女	17

内置对象

在freemarker中可以访问Request、Session等内置对象

- Request：用于获取Request对象中的attribute对象
- Session：用于获取Session对象中的attribute对象
- RequestParameters：用于获取Request对象的parameter参数（浏览器端发送的请求数据）

配置

```
1 spring:
2   freemarker:
3     #把Spring的RequestContext对象暴露为变量rc
4     request-context-attribute: rc
5     # request
6     expose-request-attributes: true
7     # session
8     expose-session-attributes: true
```

全部配置

```
1 spring:
2   freemarker:
3     # 后缀名
4     suffix: .ftl
5     content-type: text/html
6     enabled: true
7     # 缓存配置
8     cache: true
9     # 模板加载路径 按需配置（这里就是我们上面的忽略配置）
10    template-loader-path: classpath:/templates/
11    # 编码格式
12    charset: UTF-8
13    #把Spring的RequestContext对象暴露为变量rc
14    request-context-attribute: rc
15    # request
16    expose-request-attributes: true
17    # session
18    expose-session-attributes: true
19    settings:
20      # 数字格式化，无小数点
21      number_format: '0.##'
```

创建test6.ftl

```
1 <!DOCTYPE html>
2
3 <html lang="en">
4 <head>
5     <meta charset="UTF-8">
6     <title>Title</title>
7 </head>
8 <body>
9
10 <br>
11
12 请求参数key: ${RequestParameters['key']}!""
13 <br>
14 request的request1: ${Request['request1']}!""
15 <br>
16 request的request2: ${Request['request2']}!""
17 <br>
18 session的session1: ${Request['session1']}!""
19 <br>
20 session的session2: ${Request['session2']}!""
21 <br>
22 工程路径: ${rc.contextPath}
23 <br>
24 请求地址: ${rc.requestUri}
25 <br>
26
27 </body>
28 </html>
```

修改controller

```
1 package mao.spring_boot_freemarker_demo;
2
3 import mao.spring_boot_freemarker_demo.entity.Student;
4 import org.springframework.stereotype.Controller;
5 import org.springframework.web.bind.annotation.RequestMapping;
6
7 import javax.servlet.http.HttpServletRequest;
8 import javax.servlet.http.HttpSession;
9 import java.util.ArrayList;
10 import java.util.HashMap;
11 import java.util.List;
12 import java.util.Map;
13
14 /**
15  * Project name(项目名称): spring_boot_freemarker_demo
16  * Package(包名): mao.spring_boot_freemarker_demo
17  * Class(类名): TestController
18  * Author(作者): mao
19  * Author QQ: 1296193245
20  * Github: https://github.com/maomao124/
```

```

21  * Date(创建日期): 2023/1/29
22  * Time(创建时间): 19:56
23  * Version(版本): 1.0
24  * Description(描述): 无
25  */
26
27  @Controller
28  public class TestController
29  {
30      @RequestMapping("/test1")
31      public String freemarker(Map<String, Object> map)
32      {
33          map.put("name", "张三");
34          //返回模板文件名称
35          return "test1";
36      }
37
38      @RequestMapping("/test2")
39      public String freemarker2(Map<String, Object> map)
40      {
41          Student student1 = new Student();
42          student1.setId(10001L);
43          student1.setName("张三");
44          student1.setSex("男");
45          student1.setAge(18);
46
47          Student student2 = new Student();
48          student2.setId(10002L);
49          student2.setName("李四");
50          student2.setSex("男");
51          student2.setAge(19);
52
53          Student student3 = new Student();
54          student3.setId(10004L);
55          student3.setName("王五");
56          student3.setSex("女");
57          student3.setAge(17);
58
59          List<Student> list = new ArrayList<>(3);
60          list.add(student1);
61          list.add(student2);
62          list.add(student3);
63
64          map.put("stus", list);
65
66          //返回模板文件名称
67          return "test2";
68      }
69
70      @RequestMapping("/test3")
71      public String freemarker3(Map<String, Object> map)
72      {
73          Student student1 = new Student();
74          student1.setId(10001L);
75          student1.setName("张三");
76          student1.setSex("男");
77          student1.setAge(18);
78

```

```

79     Student student2 = new Student();
80     student2.setId(10002L);
81     student2.setName("李四");
82     student2.setSex("男");
83     student2.setAge(19);
84
85     Student student3 = new Student();
86     student3.setId(10004L);
87     student3.setName("王五");
88     student3.setSex("女");
89     student3.setAge(17);
90
91     Map<String, Student> studentMap = new HashMap<>();
92
93     studentMap.put("student1", student1);
94     studentMap.put("student2", student2);
95     studentMap.put("student3", student3);
96
97     map.put("studentMap", studentMap);
98
99     //返回模板文件名称
100     return "test3";
101 }
102
103 @RequestMapping("/test4")
104 public String freemarker4(Map<String, Object> map)
105 {
106     Student student1 = new Student();
107     student1.setId(10001L);
108     student1.setName("张三");
109     student1.setSex("男");
110     student1.setAge(18);
111
112     Student student2 = new Student();
113     student2.setId(10002L);
114     student2.setName("李四");
115     student2.setSex("男");
116     student2.setAge(19);
117
118     Student student3 = new Student();
119     student3.setId(10004L);
120     student3.setName("王五");
121     student3.setSex("女");
122     student3.setAge(17);
123
124     List<Student> list = new ArrayList<>(3);
125     list.add(student1);
126     list.add(student2);
127     list.add(student3);
128
129     map.put("stus", list);
130
131     //返回模板文件名称
132     return "test4";
133 }
134
135 @RequestMapping("/test5")
136 public String freemarker5(Map<String, Object> map)

```

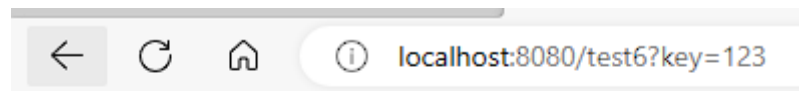
```

137     {
138         Student student1 = new Student();
139         student1.setId(10001L);
140         student1.setName("张三");
141         student1.setSex("男");
142         student1.setAge(18);
143
144         Student student2 = new Student();
145         student2.setId(10002L);
146         student2.setName("李四");
147         student2.setSex("男");
148         //一定概率为空
149         if (Math.random() > 0.5)
150         {
151             student2.setAge(19);
152         }
153         Student student3 = new Student();
154         student3.setId(10004L);
155         student3.setName("王五");
156         student3.setSex("女");
157         //一定概率为空
158         if (Math.random() > 0.5)
159         {
160             student3.setAge(17);
161         }
162
163
164         List<Student> list = new ArrayList<>(3);
165         list.add(student1);
166         list.add(student2);
167         list.add(student3);
168
169         //一定概率为空
170         if (Math.random() > 0.5)
171         {
172             map.put("stus", list);
173         }
174
175         //返回模板文件名称
176         return "test5";
177     }
178
179
180     @RequestMapping("/test6")
181     public String freemarker6(Map<String, Object> map, HttpServletRequest
182     httpServletRequest, HttpSession session)
183     {
184         //设置
185         httpServletRequest.setAttribute("request1", "1234");
186         httpServletRequest.setAttribute("request2", "5678");
187
188         session.setAttribute("session1", "1111111");
189         session.setAttribute("session2", "2222222");
190
191         //返回模板文件名称
192         return "test6";
193     }

```

运行测试

<http://localhost:8080/test6?key=123>



请求参数key: 123
request的request1: 1234
request的request2: 5678
session的session1: 1111111
session的session2: 2222222
工程路径:
请求地址: /test6

end

by mao
2023 01 29
