

Springboot关于错误页面处理和统一异常处理

01、概述

在项目访问的时候我们经常会发生错误或者页面找不到，比如：资源找不到404，服务器500错误，默认情况下springboot的处理机制都是去跳转内部的错误地址：/error 和与之对应的一个错误页面

02、项目错误页面的呈现

01、导入依赖 freemarker

```
1 <dependency>
2     <groupId>org.springframework.boot</groupId>
3     <artifactId>spring-boot-starter-
  freemarker</artifactId>
4 </dependency>
```

02、在application.yml配置

```
1 spring:
2   freemarker:
3     suffix: .html
4     cache: false
```

03、定义错误处理controller

```
1 package com.kuangstudy.web.error;
2
3 import
4   com.kuangstudy.web.error.config.MyException;
5 import org.springframework.stereotype.Controller;
6 import
7   org.springframework.web.bind.annotation.GetMapping;
8
9 import java.sql.SQLException;
10
11 /**
12  * Description:
13  * Author: yykk Administrator
14  * Version: 1.0
15  * Create Date Time: 2021/12/15 21:47.
16  * Update Date Time:
17  *
18  * @see
19  */
20 @Controller
```

```
19 public class TestErrorHandlerController {
20
21     @GetMapping("/index")
22     public String index() {
23         return "/index";
24     }
25
26
27     // 数据库错误
28     @GetMapping("/db")
29     public String db() throws SQLException {
30         if (true) {
31             throw new SQLException("数据库出错了!!!");
32         }
33         return "/db";
34     }
35
36     // 自定义异常处理
37     @GetMapping("/my")
38     public String my() {
39         if (true) {
40             throw new MyException("自定义异常!!!");
41         }
42         return "/db";
43     }
44
45
46     // 未知异常 -- 程序出现问题
47     @GetMapping("/no")
48     public String no() throws Exception {
```

```
49         if (true) {
50             throw new Exception("未知错误!!!");
51         }
52         return "/db";
53     }
54
55 }
56
```

04、定义页面index.html

```
1 <!doctype html>
2 <html lang="en">
3 <head>
4     <meta charset="UTF-8">
5     <meta name="viewport"
6         content="width=device-width, user-
7             scalable=no, initial-scale=1.0, maximum-
8             scale=1.0, minimum-scale=1.0">
9     <meta http-equiv="X-UA-Compatible"
10         content="ie=edge">
11     <title>我是首页</title>
12 </head>
13 <body>
14 <h1>我是首页</h1>
15 <div>
16     <a href="/db">处理数据库异常</a>
17     <a href="/my">处理自定义异常</a>
18     <a href="/no">处理未知错误</a>
19 </div>
```

```
17 </body>
```

```
18 </html>
```

访问：<http://localhost:8081/index>

05、证明观点

默认情况下`springboot`的出现异常的处理机制都是去跳转内部的错误地址：`/error` 和与之对应的一个错误页面

会跳转到默认`/error`对应的默认页面

← → ↻ localhost:8081/db

Whitelabel Error Page

This application has no explicit mapping for `/error`, so you are seeing this as a fallback.

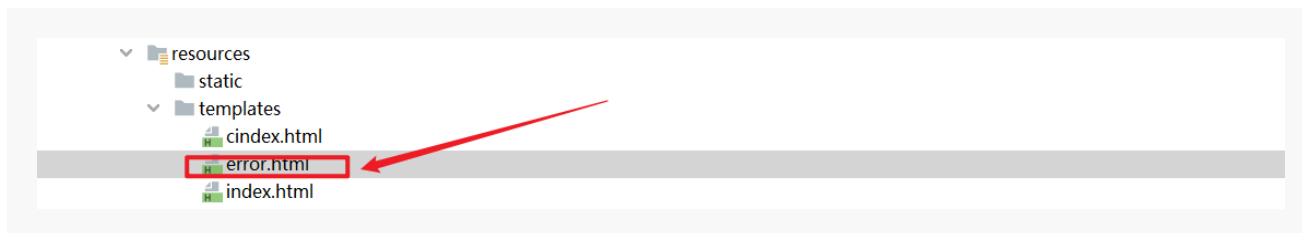
Wed Dec 15 21:53:36 CST 2021

There was an unexpected error (type=Internal Server Error, status=500).

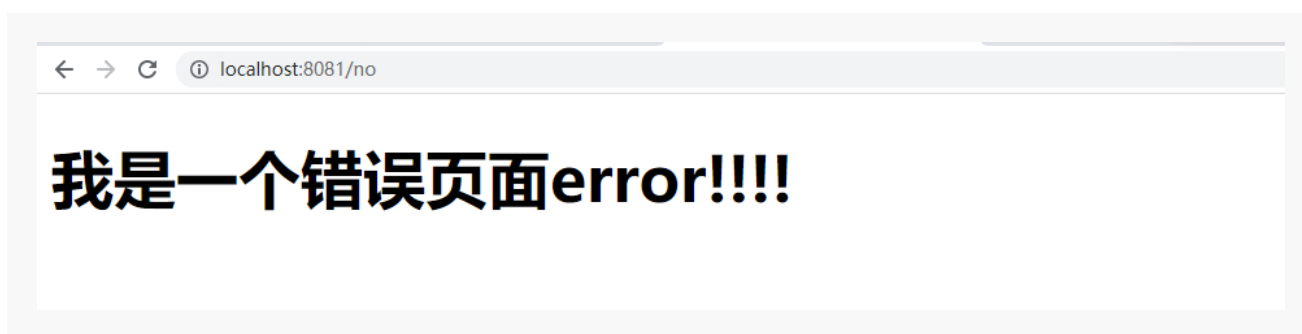
02、**Sprignboot**如何自定义错误页面

🤖 第一种方案：新建**error.html**

在/src/resources/templates新建一个error.html页面即可



访问任何一个地址，都会进入到指定的错误页面：



🤖 第二种方案：定义错误配置类

让你针对性的错误状态的错误页面，如果你没有去捕捉的异常状态，就进行全局/error

一句话：有具体的进行具体，没有具体进全局

🤖 第一步定义个错误页面的注册类，

实现接口：ErrorPageRegistrar 如下：



```
1 package com.kuangstudy.web.error.config;
2
3 import
  org.springframework.boot.web.server.ErrorPage;
4 import
  org.springframework.boot.web.server.ErrorPageRegi
    strar;
5 import
  org.springframework.boot.web.server.ErrorPageRegi
    stry;
6 import org.springframework.http.HttpStatus;
7
8 /**
9  * Description: 不推荐
10  * Author: yykk Administrator
11  * Version: 1.0
12  * Create Date Time: 2021/12/15 21:59.
13  * Update Date Time:
14  *
15  * @see
16  */
17 public class MyErrorPageRegister implements
  ErrorPageRegistrar {
18
19     @Override
20     public void
  registerErrorPages(ErrorPageRegistry registry) {
21         // 根据不同的错误状态，返回指定的错误页面
22         ErrorPage errorPage400 = new
  ErrorPage(HttpStatus.BAD_REQUEST, "/error_400");
```

```

23         ErrorPage errorPage404 = new
        ErrorPage(HttpStatus.NOT_FOUND, "/error_404");
24         ErrorPage errorPage500 = new
        ErrorPage(HttpStatus.INTERNAL_SERVER_ERROR,
        "/error_500");
25         // 注册
26         registry.addErrorPages(errorPage400,
        errorPage404, errorPage500);
27     }
28 }
29

```

👉 第二步定义个错误配置类，把**springboot**内部的错误配置类机制覆盖成自己的

```

1 package com.kuangstudy.web.error.config;
2
3 import
    org.springframework.boot.SpringBootConfiguration;
4 import
    org.springframework.boot.web.server.ErrorPage;
5 import
    org.springframework.boot.web.server.ErrorPageRegi
    star;
6 import
    org.springframework.boot.web.server.ErrorPageRegi
    stry;
7 import
    org.springframework.context.annotation.Bean;
8 import org.springframework.http.HttpStatus;

```



```

9
10 /**
11  * Description: 不推荐
12  * Author: yykk Administrator
13  * Version: 1.0
14  * Create Date Time: 2021/12/15 21:59.
15  * Update Date Time:
16  *
17  * @see
18  */
19 @SpringBootConfiguration
20 public class ErrorPageRegisterConfiguration {
21
22     @Bean
23     public ErrorPageRegistrar
24     errorPageRegistrar(){
25         return new MyErrorPageRegister();
26     }
27 }

```

 第三步：定义每个错误页面跳转的地址

```

1 package com.kuangstudy.web.error.config;
2
3 import org.springframework.stereotype.Controller;

```

```
4 import
  org.springframework.web.bind.annotation.GetMapping;
5
6 /**
7  * Description:
8  * Author: yykk Administrator
9  * Version: 1.0
10 * Create Date Time: 2021/12/15 22:04.
11 * Update Date Time:
12 *
13 * @see
14 */
15 @Controller
16 public class ErrorController {
17
18     @GetMapping("/error_400")
19     public String errorpage1() {
20         return "400";
21     }
22
23     @GetMapping("/error_404")
24     public String errorpage2() {
25         return "404";
26     }
27
28     @GetMapping("/error_500")
29     public String errorpage3() {
30         return "500";
31     }
32 }
```

简化如下：

```
1 package com.kuangstudy.web.error.config;
2
3 import org.springframework.stereotype.Controller;
4 import
  org.springframework.web.bind.annotation.GetMapping;
5 import
  org.springframework.web.bind.annotation.PathVariable;
6
7 /**
8
9  * Description:
10 * Author: yykk Administrator
11 * Version: 1.0
12 * Create Date Time: 2021/12/15 22:04.
13 * Update Date Time:
14 * @see
15 */
16 @Controller
17 public class ErrorController {
18
19     @GetMapping("/error_{status}")
20     public String
  errorpage1(@PathVariable("status") String status)
  {
21         return status;
22     }
23 }
```

第四步：访问错误地址：

<http://localhost:8081/db----500> --- 几乎全部进行500错误

<http://localhost:8081/indexxxx> ---404----404错误页面

<http://localhost:8081/db> ----403----定义是post---访问是get -----
出现/error

总结：

其实在开发中，其实没必要这么明细，越明细越好，但是会增大你的工作量。有时候直接给用户一个统一的错误就可以了。

第三种方式：自定义错误页面的名字

```
1 package com.kuangstudy.web.error;  
2  
3 import  
    org.springframework.boot.web.servlet.error.ErrorC  
    ontroller;  
4 import org.springframework.stereotype.Controller;  
5 import  
    org.springframework.web.bind.annotation.GetMappin  
    g;  
6  
7 /**  
8  * Description:
```

```
9  * Author: yykk Administrator
10 * Version: 1.0
11 * Create Date Time: 2021/12/15 22:16.
12 * Update Date Time:
13 *
14 * @see
15 */
16 @Controller
17 public class MyErrorPageController implements
    ErrorController {
18
19     @GetMapping("/error")
20     public String errorindex() {
21         return "/errorpage";
22     }
23 }
24
```

- 这种方式其实是对错误页面的自定义方式，仅此而已。如果你对默认规则是明白的话。这种其实意义不大。
- 好处：我可以去控制，错误页面的目录，方便隔离和控制
- 上面第一种坏处就是：error.html必须放在templates下方