7、Web-错误处理机制

2020年6月9日 10:56

- 1、spring-boot的默认错误处理机制
  默认效果：
  - 1）浏览器，返回一个默认的错误界面
  - 2）其他客户端返回json数据
- 2、如何定制错误响应
  - 1）如何定制错误页面
    - 1）有模板引擎的情况下；error/状态码；【将错误页面命名为 错误状态码.html 放在模板引擎文件夹里面的 error文件夹下】，发生此状态码的错误就会来到 对应的页面；
      我们可以使用4xx和5xx作为错误页面的文件名来匹配这种类型的所有错误，精确优先（优先寻找精确的状态码.html）；
      页面能获取的信息；
      timestamp：时间戳
      status：状态码
      error：错误提示
      exception：异常对象
      message：异常消息
      errors：JSR303数据校验的错误都在这里
    - 2）、没有模板引擎（模板引擎找不到这个错误页面），静态资源文件夹下找；
    - 3）、以上都没有错误页面，就是默认来到SpringBoot默认的错误提示页面；
  - 2）如何定制错误的json数据
    - 1）自定义异常处理&返回定制json数据

```
@ResponseBody
@ExceptionHandler(UserNotExistException.class)
publicMap<String,Object>handlerException(Exceptione){
Map<String,Object>map=newHashMap<>();
map.put("code","user.notexist");
map.put("message",e.getMessage());
returnmap;
}
//没有自适应效果
```

    - 2）转发到/error进行自适应响应效果处理

```
@ExceptionHandler(UserNotExistException.class)
publicStringhandlerException(Exceptione,HttpServletRequestrequest){

Map<String,Object>map=newHashMap<>();
map.put("code","user.notexist");

map.put("message","用户出错了");
request.setAttribute("ext",map);
return"forward:/error";
```

```
}
```

- 3）将我们的定制数据携带出去

  出现错误以后，会来到/error请求，会被BasicErrorController处理，响应出去可以获取的数据是由getErrorAttributes得到的（是AbstractErrorController（ErrorController）规定的方法）；

  - 1、完全来编写一个ErrorController的实现类【或者是编写AbstractErrorController的子类】，放在容器中；
  - 2、页面上能用的数据，或者是json返回能用的数据都是通过errorAttributes.getErrorAttributes得到；

    容器中DefaultErrorAttributes.getErrorAttributes()；默认进行数据处理的；

    自定义ErrorAttributes

```
publicMyErrorAttributes(){
super(true);
}

@Override
publicMap<String,Object>
getErrorAttributes(WebRequestwebRequest,booleanincludeStackTrace){

Map<String,Object>ext=(Map<String,Object>)
webRequest.getAttribute("ext",0);
Map<String,Object>
errorAttributes=super.getErrorAttributes(webRequest,includeStackTrace);
errorAttributes.put("company","atLhq");
errorAttributes.put("ext",ext);



returnerrorAttributes;

}
```