

一、功能说明

工具包基于过滤器->拦截器->切面编程规范设计, 对响应的Object和Model数据进行脱敏处理。

背景

- 随着平台业务越发广泛用户资料积累日益激增, 其信息保护势在必行。
- 应国家政策及平台安全组要求, 统一对接口响应数据进行脱敏处理。

二、工具使用

- 添加依赖: pom.xml

```
<dependency>
  <groupId>cn.gmlee</groupId>
  <artifactId>tools-request</artifactId>
  <version>3.1.0</version>
</dependency>
```

- 启用工具

在启动类上或其他配置类上扫描包: cn.gmlee.tools

```
@SpringBootApplication
@ComponentScan("cn.gmlee.tools")
public class XxxApp {
    public static void main(String[] args) {
        SpringApplication.run(XxxApp.class, args);
    }
}
```

- 脱敏配置: application.properties

```
# use: 默认true表示使用正则匹配方式脱敏
# use: false表示rule是属性名称发现该属性则将其值进行脱敏
# 身份证参考: \\d{15}|\\d{18}
# 手机号码参考: \\d{3}-\\d{8}|\\d{4}-\\d{7}
# 电子邮箱参考: \\w+([-+.]\\w+)*@\\w+([-+.]\\w+)*\\.\\w+([-+.]\\w+)*
# 手机号参考: [1-9]\\d{5}(?!\\d)
tools.request.desensitization[0].rule=\\d{15}|\\d{18}
tools.request.desensitization[1].rule=\\d{3}-\\d{8}|\\d{4}-\\d{7}
tools.request.desensitization[2].rule=\\w+([-+.]\\w+)*@\\w+
([-+.]\\w+)*\\.\\w+([-+.]\\w+)*
tools.request.desensitization[3].rule=[1-9]\\d{5}(?!\\d)
tools.request.desensitization[4].rule=1(3[0-9]|5[0-3,5-9]|7[1-3,5-8]|8[0-9])\\d{8}
tools.request.desensitization[5].rule=name
tools.request.desensitization[5].use=false
#
tools.request.desensitizationClassExcludes=com.gm.demo.tools.request.controller.vo.Demo
```

三、工具原理

逻辑

- 通过切面拦截返回值
- 遍历其属性名 或 值判断是否符合脱敏规则
- 脱敏后将脱敏的数据重新覆盖值

代码

1. 处理对象数据: handler(Object obj)
2. 脱敏对象: desensitizationFor(Object obj)
3. 配置说明

```
/**
 * @author Jas°
 * @date 2021/4/29 (周四)
 */
@Data
public class Desensitization implements Serializable {
    /**
     * 是否在规则中使用正则表达式
     * false: 使用属性名称为依据判断是否对值脱敏
     */
    private Boolean use = true;
    /**
     * 规则: 正则表达式 或 属性名称
     */
    private String rule;
}
```

数据

- 由配置加载到运行内存中

四、使用示例

4.1 案例演示

- 响应对象

```
import lombok.Data;

import java.util.List;
import java.util.Map;

/**
```

```

* 示范实体
* @author Jas°
* @date 2021/4/25 (周日)
*/
@Data
public class Demo {
    private Long id;
    private String name;
    private String number;
    private List<Demo> demos;
    private List<String> numbers;
    private List<Map<String, String>> maps;
}

```

- JSON响应示例接口

```

@ResponseBody
@RequestMapping("test2")
public JsonResult<Demo> test2(@RequestBody Demo demo) throws Exception {
    return JsonResult.OK.newly(demo);
}

```

- 普通视图响应示例接口

```

@RequestMapping("test1")
public ModelAndView test1(Demo demo) throws Exception {
    ModelAndView mv = new ModelAndView();
    mv.setViewName("demo");
    mv.addObject("demo", demo);
    return mv;
}

```

- 脱敏配置

```

tools.request.desensitization[0].rule=\\d{15}|\\d{18}
tools.request.desensitization[1].rule=\\d{3}-\\d{8}|\\d{4}-\\d{7}
tools.request.desensitization[2].rule=\\w+([-+.]\\w+)*@\\w+
([-.]\\w+)*\\.\\w+([-.]\\w+)*
tools.request.desensitization[3].rule=[1-9]\\d{5}(?!\\d)
tools.request.desensitization[4].rule=1(3[0-9]|5[0-3,5-9]|7[1-3,5-8]|8[0-9])\\d{8}
tools.request.desensitization[5].rule=name
tools.request.desensitization[5].use=false
# 如果调试中发现特殊类需要排除可指定一下配置
# 温馨提示：该配置仅用于：非JSON响应
#
tools.request.desensitizationClassExcludes=com.gm.demo.tools.request.controller.vo.Demo

```

- JSON响应示例

POST http://localhost:8080/test2

Params Authorization Headers (9) Body Pre-request Script Tests Settings

none form-data x-www-form-urlencoded raw binary GraphQL JSON

```
1 {
2   "id": 110,
3   "name": "你好",
4   "numbez": "1376766337"
5 }
```

Body Cookies Headers (5) Test Results

Pretty Raw Preview Visualize JSON

```
1 {
2   "code": 1000,
3   "msg": "操作成功",
4   "data": {
5     "id": 110,
6     "name": "你*",
7     "numbez": "137*****237"
8   },
9   "responseTime": "1619691103424"
10 }
```

```
{
  "code": 1000,
  "msg": "操作成功",
  "data": {
    "id": 110,
    "name": "你*",
    "number": "137*****237"
  },
  "responseTime": "1619693036508"
}
```

- 普通视图响应示例

POST http://localhost:8080/test1

Params Authorization Headers (9) Body Pre-request Script Tests Settings

none form-data x-www-form-urlencoded raw binary GraphQL

KEY	VALUE	DESCRIPTION	...	Bulk Edit
<input checked="" type="checkbox"/> id	110			
<input checked="" type="checkbox"/> name	你好			
<input checked="" type="checkbox"/> demos[0].name	哈喽			
<input checked="" type="checkbox"/> numbers[0]	112			
<input checked="" type="checkbox"/> maps[0][name]	哟哟			

Body Cookies Headers (6) Test Results

Pretty Raw Preview Visualize HTML

```
1 <!DOCTYPE html>
2 <html lang="en">
3
4 <head>
5   <meta charset="UTF-8">
6   <title>Title</title>
7 </head>
8
9 <body>
10   <input>110</input>
11   <input>你*</input>
12   <input>哈*</input>
13   <input>112</input>
14   <input>哟*</input>
15 </body>
16
17 </html>
```

```
<!DOCTYPE html>
<html lang="en">

<head>
  <meta charset="UTF-8">
  <title>Title</title>
</head>

<body>
  <input>110</input>
  <input>你*</input>
  <input>哈*</input>
  <input>112</input>
  <input>哟*</input>
```

```
</body>
```

```
</html>
```

4.2 实验室源码

- [tools-request](#)