

一、功能说明

该工具包的作用是防止用户误点击, 造成的重复请求, 致使后端和数据库压力增加。

背景

- OMS公海线索搜索数据压力大, 单次请求延迟在3s以上
- 用户在点击等待当中, 前端并未友好限制或提示用户已经提交请求需要等待, 导致用户重复点击

二、工具使用

- 添加依赖: pom.xml

```
<dependency>
  <groupId>cn.gmlee</groupId>
  <artifactId>tools-api</artifactId>
  <version>3.0.0</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);
    }
}
```

- 添加注解: @Once

```
@Once
@RequestMapping("...")
public JsonResult come(Object... os) {
    return JsonResult.OK;
}
```

- 注解说明: @Once(3000)
 - 默认值: 3000
 - 单位: 毫秒
 - 特别说明: Request请求头中携带: token (不传token表示: 所有用户在指定时间内不能重复访问)

三、工具原理

逻辑

- 该工具通过AOP实现
- 每次请求访问注解加持的方法时将预先检测是否在指定时间范围内已经访问过
- 如果已经访问过则抛出异常 (请注意扑捉异常, 异常码: 202, 异常信息: 请求频繁)

代码

1. 获取当前请求的请求头中的token和url?QueryString (以下称: 接口)
2. 根据接口获取所有用户访问数据 (包含下一次可请求的时间)
3. 判断当前时间与下一次可请求时间是否符合逻辑
4. 符合逻辑则允许请求, 否则抛出异常: 请求频繁

数据

- 访问记录存储在应用的内存当中 (以下称: Once数据)
- 请求一次@Once接口清理一次Once数据

四、使用示例

- 首次访问

```
{
  "code": 200,
  "msg": "操作成功",
  "data": null,
  "responseTime": 1604403816747
}
```

- 3s内再次访问

```
{
  "code": 202,
  "msg": "请求频繁",
  "data": null,
  "responseTime": 1604403897217
}
```

- 3s后再次访问

```
{
  "code": 200,
  "msg": "操作成功",
  "data": null,
  "responseTime": 1604403816747
}
```

