

Thymeleaf中SpringSecurity的使用

Spring Security 可以在一些视图技术中进行控制显示效果。例如：`JSP` 或 `Thymeleaf`。在非前后端分离且使用 Spring Boot 的项目中多使用 `Thymeleaf` 作为视图展示技术。

Thymeleaf 对 Spring Security 的支持都放在 `thymeleaf-extras-springsecurityx` 中，目前最新版本为 5。所以需要在项目中添加此 jar 包的依赖和 `thymeleaf` 的依赖。。

```
<!--thymeleaf springsecurity5 依赖-->
<dependency>
  <groupId>org.thymeleaf.extras</groupId>
  <artifactId>thymeleaf-extras-springsecurity5</artifactId>
</dependency>
<!--thymeleaf依赖-->
<dependency>
  <groupId>org.springframework.boot</groupId>
  <artifactId>spring-boot-starter-thymeleaf</artifactId>
</dependency>
```

在 html 页面中引入 `thymeleaf` 命名空间和 `security` 命名空间

```
<html xmlns="http://www.w3.org/1999/xhtml"
      xmlns:th="http://www.thymeleaf.org"
      xmlns:sec="http://www.thymeleaf.org/thymeleaf-extras-springsecurity5">
```

获取属性

可以在html页面中通过 `sec:authentication=` 获取

`UsernamePasswordAuthenticationToken` 中所有 `getxxx` 的内容，包含父类中的 `getxxx` 的内容。

根据源码得出下面属性：

- `name`：登录账号名称
- `principal`：登录主体，在自定义登录逻辑中是 `UserDetails`
- `credentials`：凭证
- `authorities`：权限和角色
- `details`：实际上是 `WebAuthenticationDetails` 的实例。可以获取 `remoteAddress` (客户端 ip) 和 `sessionId` (当前 sessionId)

新建demo.html

在项目 `resources` 中新建 `templates` 文件夹，在 `templates` 中新建 `demo.html` 页面

```
<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml"
      xmlns:sec="http://www.thymeleaf.org/thymeleaf-extras-springsecurity5">
<head>
  <meta charset="UTF-8">
```



```
<title>Title</title>
</head>
<body>
  登录账号:<span sec:authentication="name"></span><br/>
  登录账号:<span sec:authentication="principal.username"></span><br/>
  凭证: <span sec:authentication="credentials"></span><br/>
  权限和角色: <span sec:authentication="authorities"></span><br/>
  客户端地址: <span sec:authentication="details.remoteAddress"></span><br/>
  sessionId: <span sec:authentication="details.sessionId"></span><br/>
</body>
</html>
```

编写Controller

thymeleaf 页面需要控制转发，在控制器类中编写下面方法

```
@RequestMapping("/demo")
public String demo(){
    return "demo";
}
```

