Spring方式配置spring sucurity

Maven依赖

<dependency>

<groupId>org.springframework.security</groupId>

<artifactId>spring-security-taglibs</artifactId>

<version>5.0.5.RELEASE</version>

</dependency>

<dependency>

<groupId>org.springframework.security</groupId>

<artifactId>spring-security-web</artifactId>

<version>5.0.5.RELEASE</version>

</dependency>

<dependency>

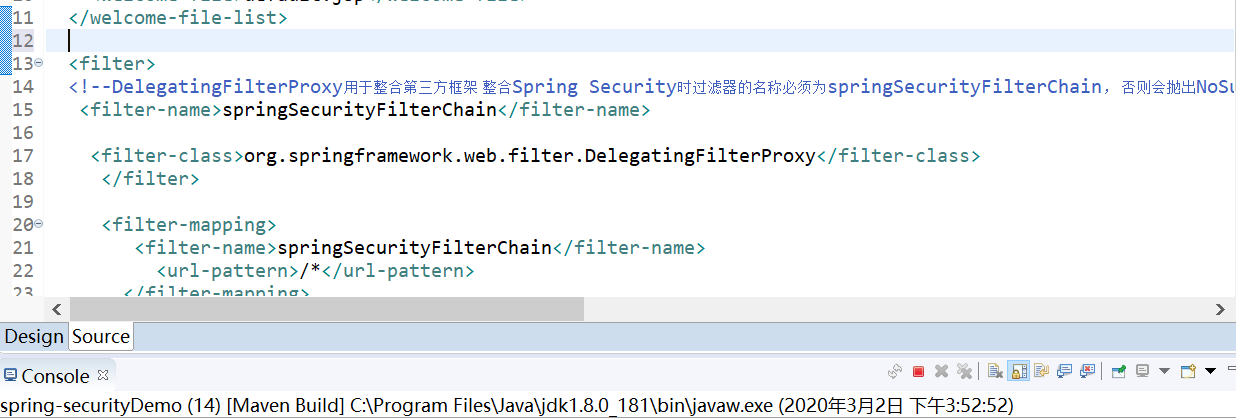
<groupId>org.springframework.security</groupId>

<artifactId>spring-security-config</artifactId>

<version>5.0.5.RELEASE</version>

</dependency>

第一步配置过滤器 名字固定



请求成功以后 自动配置



拦截所有请求 验证有没有自定义角色 默认跳转默认登录页面

Login-processing-url 登录的请求路径 必须+/

Default-target-url 登录成功以后跳的请求

Authentication-failure-url 登录失败跳的页面

必须开启 报403就是没开启 被禁用

<security:csre disabled=”true”/>

<!--authentication-manager：认证管理器，用于处理认证操作 -->

<security:authentication-manager> <!--authentication-provider：认证提供者，执行具体的认证逻辑 -->

<security:authentication-provider> <!--user-service：用于获取用户信息，提供给authentication-provider进行认证 -->

<security:user-service> <!--user：定义用户信息，可以指定用户名、密码、角色，后期可以改为从数据库查询 用户信息 {noop}：表示当前使用的密码不加密 authorities="ROLE\_ADMIN"

表示当前用户授哪个角色 -->

<security:user name=*"admin"* password=*"{noop}admin"*

authorities=*"ROLE\_ADMIN"*></security:user>

</security:user-service>

</security:authentication-provider>

</security:authentication-manager>

配置对密码加密 用户名密码数据库封装

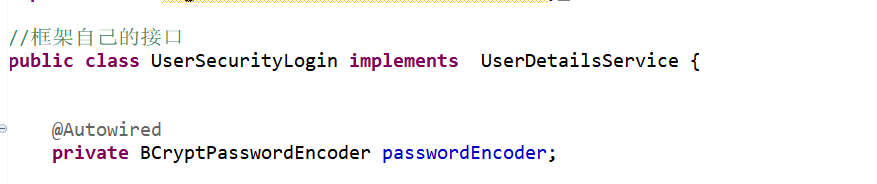
每次传用户名数据库查询 框架自动跟查到的密码跟页面密码进行比较 并且给权限

<bean id=*"userService"* class=*"com.test.servicelmpl.UserSecurityLogin"*></bean>

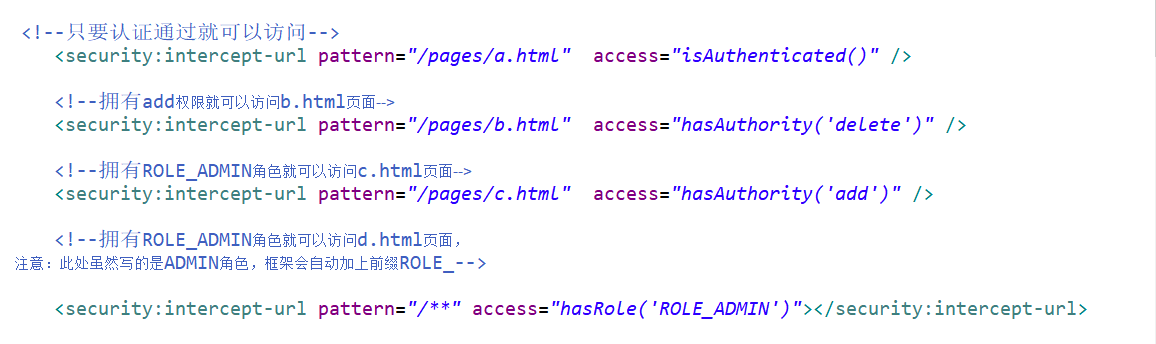
<!-- 对密码加密 -->

<bean id=*"passwordEncoder"* class=*"org.springframework.security.crypto.bcrypt.BCryptPasswordEncoder"*/>









配置访问的东西 需要什么角色跟权限