问题1

thymeleaf

<!--没有结束符的HTML5标签解析出错的解决办法 -->

<!-- 配置spring.thymeleaf.mode =LEGACYHTML5 -->

<!--LEGACYHTML5需要搭配一个额外的库NekoHTML才可用 -->

<groupId>net.sourceforge.nekohtml</groupId>

<artifactId>nekohtml</artifactId>

<version>1.9.22</version>

问题2

spring Security 使用时禁止了frame

httpSecurity.headers().frameOptions().sameOrigin();//解决前端使用layui。tab嵌套iframe,在加入security之前并无异常

//springmvc 在整合spring security时，浏览器发送请求

//Refused to display 'url' in a frame because it set 'X-Frame-Options' to 'deny'.

问题3

无法加载frame

layUI在js中设置的路径，需要修改。

然后springboot 在templates中放HTML，需要通过controller访问。

放在static下，则可以直接访问了。

而springboot默认static文件夹之外的页面需要通过controller访问（好像是websecurity权限控制）。

可以使用viewController 解决

<https://blog.csdn.net/nerversimply/article/details/79575115>

问题4

LayUI

index.html

后台管理系统

左边导航栏，预留一个div

然后在加载整个项目的时候，js自加载按钮信息，填充左边菜单栏

菜单栏的标题和连接，都可以写在一个js数据中，然后获取填充。

问题5

spring security

th:src和@{}这都是什么鬼。其实这是Thymeleaf的语法。@{}是引入外部资源用的。

th:action 可以自动加入crsf属性

在使用spring security时，若用Thymeleaf模板

指定 login.html为登录界面

login.html 中登录时，提交表单时 action 用 th:acton = "@{/login}".

退出也要使用

@{/logout}

这是 thymeleaf 和spring security 组合使用的写法。

问题6.

springboot 的静态资源问题

当我们访问地址 http://localhost:8080/fengjing.jpg 的时候，显示哪张图片？

优先级顺序为：这些目录到在src/maim/resources/ 下 META/resources > resources > static > public

通过配置文件配置

上面是使用代码来定义静态资源的映射，其实Spring Boot也为我们提供了可以直接在 application.properties（或.yml）中配置的方法。

配置方法如下：

来定义静态资源的映射，其实Spring Boot也为我们提供了可以直接在 application.properties（或.yml）中配置的方法。

配置方法如下：

# 默认值为 /\*\*

spring.mvc.static-path-pattern=

# 默认值为 classpath:/META-INF/resources/,classpath:/resources/,classpath:/static/,classpath:/public/

spring.resources.static-locations=这里设置要指向的路径，多个使用英文逗号隔开，

使用 spring.mvc.static-path-pattern 可以重新定义pattern，如修改为 /myres/\*\* ，则访问static 等目录下的fengjing.jpg文件应该为 http://localhost:8080/myres/fengjing.jpg ，修改之前为 http://localhost:8080/fengjing.jpg

使用 spring.resources.static-locations 可以重新定义 pattern 所指向的路径，支持 classpath: 和 file:

问题7

如果不小心引入其他jar包，

可能在项目启动的时候报错，原因是自动配置的时候在

classpath中找不到对应的bean。

问题8

springmvc 处理 httpclient post请求 ,可以访问本项目，但是访问不到？

springboot使用了安全验证 ，如果发送post请求数据，那么 很多问题都应该是springboot框架带来的问题。

Spring Boot 使用 Spring Security 后无法 POST 提交数据解决方法

【问题描述】

项目使用的是 spring-boot + spring-security，页面用了 thymeleaf 模板

使用了 spring-security 后，默认开启了防止跨域攻击的功能，任何 POST 提交到后台的表单都要验证是否带有 \_csrf 参数，一旦传来的 \_csrf 参数不正确，服务器便返回 403 错误；

Invalid CSRF Token 'null' was found on the request parameter '\_csrf' or header 'X-CSRF-TOKEN'.

解决方法一：form 表单中添加 \_csrf 隐藏域

<input type="hidden" th:name="${\_csrf.parameterName}" th:value="${\_csrf.token}"/>

做为提交到后台的 \_csrf 值

解决方法二：form 表单使用 th:action 属性， thymeleaf 会自动在 form 表单中生成 \_csrf 隐藏域；

th:action="@{/login}"

解决方法三：关闭防跨域攻击功能，使用 http.csrf().disable()：

【总结】

个人比较推荐方法二；

方法一需要手动在 form 表单中添加 \_csrf 隐藏域，比较麻烦，个人并不怎么推荐；

方法三有点儿简单粗暴，如果网站没必要防止跨域攻击的，可以直接用方法三；

9．Unknown column aa' in 'where clause'

Where 子句中 列有问题，一般情况下是引用的列没有加引号所致。

10. No property matchId found for type PpInjectImage!

jpa中自定义方法，写的查询条件，对应的entity中没有方法中的属性。

11.

## 一、什么是跨域访问

**举个栗子**：在A网站中，我们希望使用Ajax来获得B网站中的特定内容。如果A网站与B网站不在同一个域中，那么就出现了跨域访问问题。你可以理解为两个域名之间不能跨过域名来发送请求或者请求数据，否则就是不安全的。跨域访问违反了同源策略，同源策略的详细信息可以点击如下链接：[Same-origin\_policy](https://en.wikipedia.org/wiki/Same-origin_policy)；   
总而言之，同源策略规定，浏览器的ajax只能访问跟它的HTML页面同源（相同域名或IP）的资源。

## 二、解决方案

常用的解决方案有两种，可以分为客户端解决方案和服务器端解决方案。先说服务器端解决方案：

* 服务器端解决方案   
  在服务器端的filter或者servlet里面添加   
  response.setHeader("Access-Control-Allow-Origin", "\*");   
  “Access-Control-Allow-Origin”表示允许跨域访问，“\*”表示允许所有来源进行跨域访问，这里也可以替换为特定的域名或ip。   
  很显然，这种方式对非网站拥有人员来说是不能做到的。而且此种方式很容易受到CSRF攻击。
* 客户端解决方案

1. $(function($){
2. var url = 'http://\*\*\*\*\*/index';
3. $.ajax(url, {
4. data: {
5. 'cityname': '成都',
6. 'date': '2016.12.12'
7. },
8. datatype: 'jsonp',
9. crossDomain: true,
10. success: function(data) {
11. if(data && data.resultcode == '200'){
12. console.log(data.result.today);
13. }
14. }
15. });

将ajax请求中的dataType属性设置为“jsonp”，jsonp是专门用来解决跨域访问而诞生的。

12. 使用第三方JS，要注意引入的顺序，不然会识别不了，报错，识别不了该方法。

在其他的都正确的时候，要注意是不是JS的引入顺序导致的。

13.未解决问题

1.打包部署springboot项目失败

2.logging自定义不行

3.diagester 解析失败