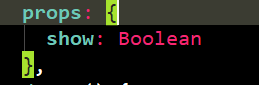
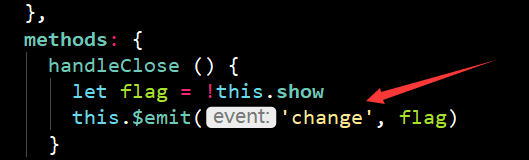
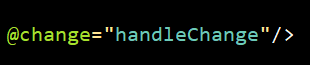
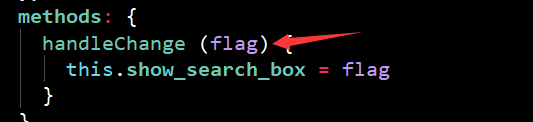
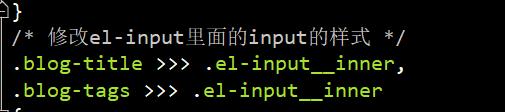
前端方面:

1. **父子组件的信息传递**

解决: 通过$emit的方式，父组件通过绑定一个属性, 子组件通过获取这个属性, 并且设置数据类型, 子组件通过自身的一个函数去用$emit的方式去触发另一个函数, 父组件监听这个函数的方式, 子组件$emit触发函数的第二个参数, 就是父组件监听函数的参数

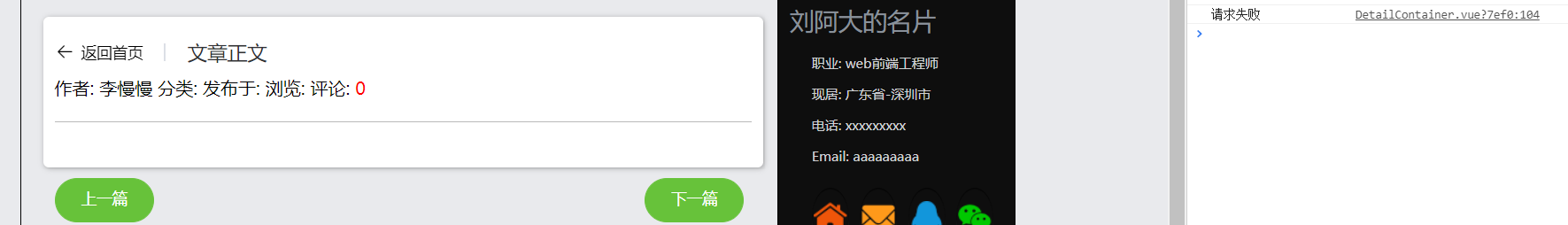
1. **前期vue-cli构建脚手架的时候, 如果选择了Eslint的话, 前期在编辑器编写代码的时候会出现很多波浪线的代码不规范的警告, 以及在控制台调试代码的时候会报很多代码不规范的警告, 可以直接在编辑器右键选择eslint, 也可以在config文件夹下的index.js里面禁用eslint**
2. **使用element-ui组件时, 无法修改el-inpu组件里的input**

解决: 在选中el-input的类名的同时用>>>选中里面的input元素

1. **npm run build打包之后, 样式出现问题**

解决: 原因是打包之后, 样式会压缩. 所以最好在一开始设置样式的时候, 给style加上一个scoped的属性限制一下作用域. 也可以使用less语句. 用了less还是有一点点位置的偏差, 是因为文本标签自带的margin没有初始化0

1. **页面刷新后, 数据会不存在(测试中出现的问题)**



原因: 写在data里的数据是放在内存里的, 刷新后会消失, 如果想继续使用的话, 就得做持久化的处理.

解决: 在刷新页面之前, 将数据存放到sessionStorage中

sessionStorage.setItem('blog', JSON.stringify(this.**blog**));

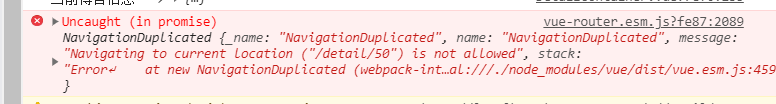
然后在实例created钩子函数中再取出来

created: function() {  
 if(sessionStorage.getItem('blog')) {  
 let blog = JSON.parse(sessionStorage.getItem('blog'))  
 this.**blog** = blog;  
 }

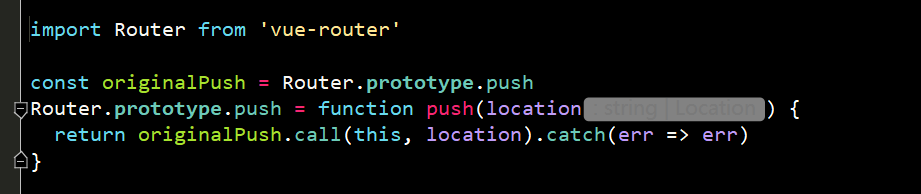
验证码需要用到的包



vue的router-link报错，两个按钮的router-link对应的组件是一个，不同的按钮请求的数据不同，每次进行刷新会报这个错误，这个错误却不影响使用；



解决: 在main.js 加入以下代码



后台服务器方面: