1. **项目结构说明**
2. src文件下的目录说明

|  |  |
| --- | --- |
| **目录名** | **目录分类说明** |
| api | Api接口请求放在该目录 |
| assets | 静态资源放在该目录 |
| components | 公共组件放在该目录（注：页面组件不要放在此目录） |
| config | 构建项目配置目录 |
| directive | 自定义全局指令放在该目录 |
| filters | 自定义全局过滤器放在该目录 |
| libs | 公共库、全局方法及全局函数放在该目录 |
| locale | 本地设置语言配置目录 |
| mock | 模拟数据axios请求api目录 |
| plugin | 插件目录，所有非npm包管理安装的插件放在该目录 |
| router | 配置项目路由目录 |
| store | 全局状态管理目录 |
| view | 所有页面及页面组件放在该目录 |

1. src目录下具体文件夹说明及用法

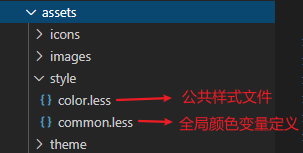
（1）.api目录下文件说明



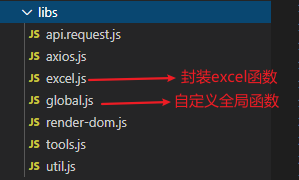
图中蓝色文字表示五大模块的统一管理的api方法，红色文字表示公共及其他等统一管理的api方法

1. .assets目录下说明

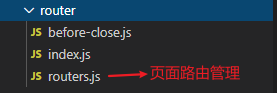
element-ui全局样式，默认size为small

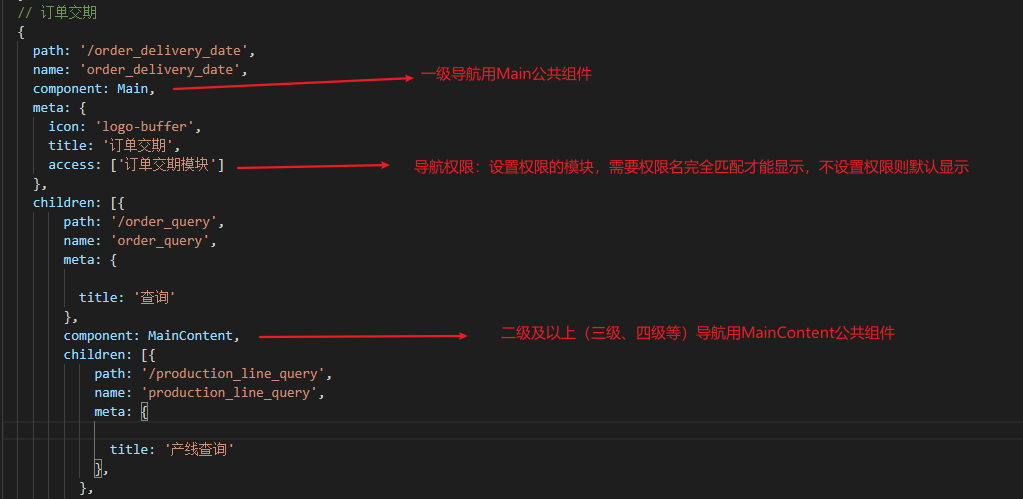


1. lib目录下说明



1. router目录下说明





注：路由中，path可以随便设置路径，但是name必须是唯一的且与页面组件的name一致，

页面导航跳转统一用name来跳转，不要使用path跳转，因为path修改路径变动大。

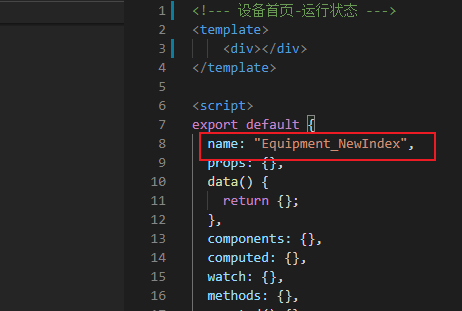
 this.$router.push({

        name: '页面组件名字'

      });

例如：

Equipment-NewIndex.vue



routers.js



1. views目录下文件说明

按路由导航层级建文件夹及命名



二．公共组件使用

1. 页面内导航公共组件用法说明

引入：import SecondNav from "\_c/ second-nav";

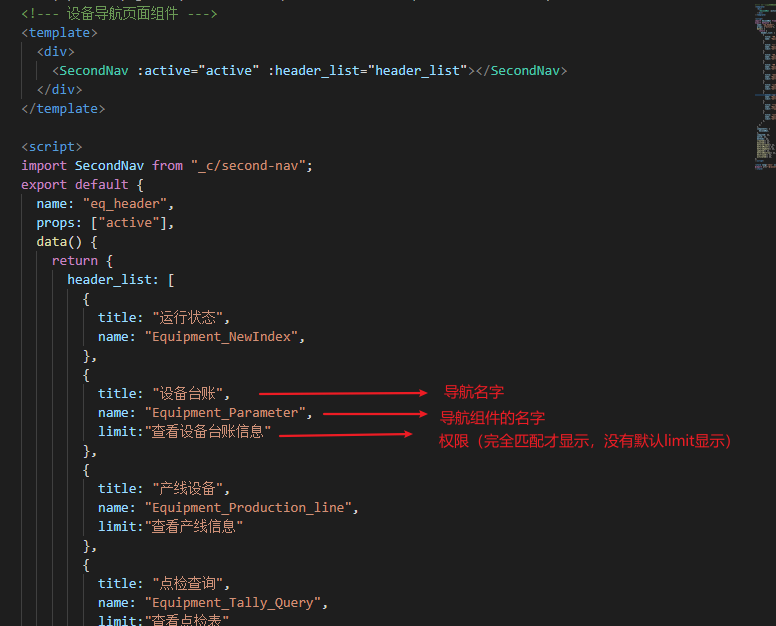
名字：<SecondNav></SecondNav>

active（string）：选中的导航名

header\_list（array）：导航信息，【包括title，name，limit】



例如：



1. 下载模板按钮公共组件用法说明

引入：import DownloadTemplate from "\_c/download-template";

名字：<DownloadTemplate></DownloadTemplate>

templateName（string）：下载模板文件的名称（带后缀）

url（string）：下载模板文件的路径



例如：



三．直接调用

this.$userInfo：用户信息

this.$limit(‘权限名’)：权限判断

this.$loadPath：路径上传下载（判断是否生产/开发环境及内/外网）