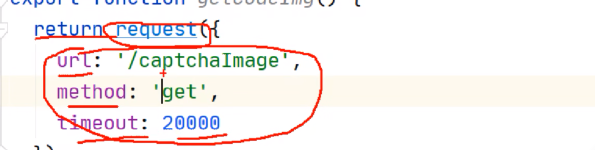
验证码的前端实现

嵌套：

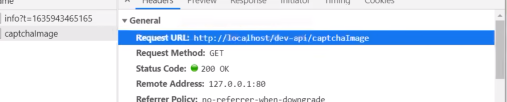


3步封装：url/method/timeout



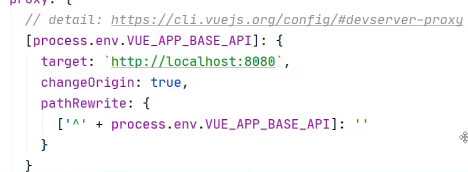
/开发环境 /dev-api（前缀）

/开发路径（名字特殊）：



Vue获取图片：后端 后端端口：8080

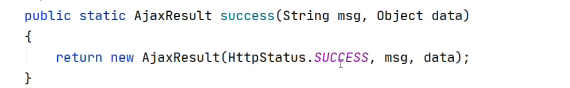
反向代理：url请求前端，进行代理，映射到后端，解决跨越问题



把 dev-api 替换成’ ’，再映射到 http：//localhost：8080

验证码的后端实现

/CaptchaController



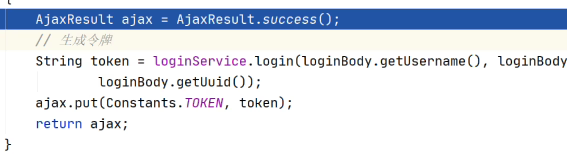
前后端交互处



分割： 4+1 =?|@5 |（分割）

ajax 实现前端和后端的交互

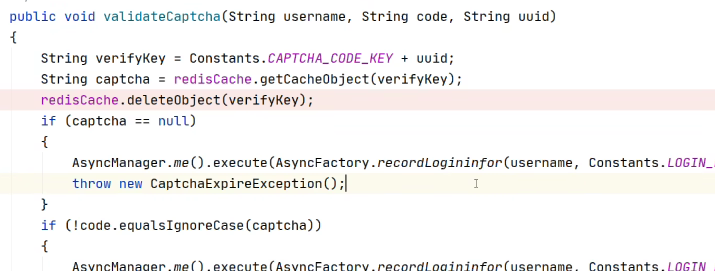
生成令牌：



登录的实现

后端： 校验验证码-校验用户名和密码=生成Token

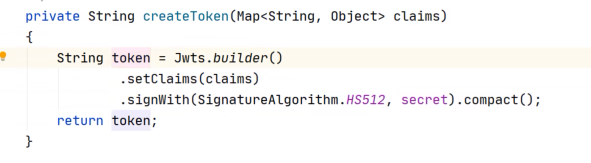
异步代码，不影响其他进程



生成token：







JWt：加密，生成字符串/token，返回前端，完成整个过程

登录的具体流程：校验验证码-校验用户名和密码-生成token

使用异步任务管理器，结合线程池，实现了异步的操作日志和业务逻辑实现异步耦合。

Getlnfo:获取用户信息

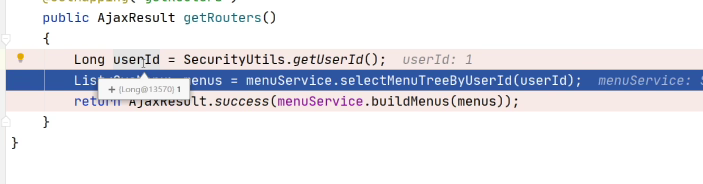


获取当前用户的角色和权限信息，存储到Vuex中

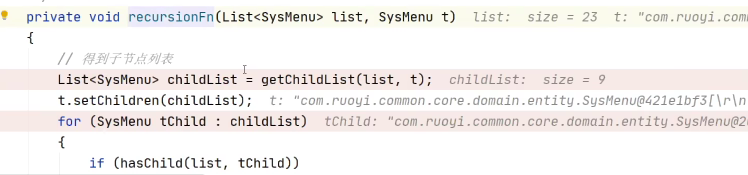
\*：\*：\* 所有的权限

###getRouters

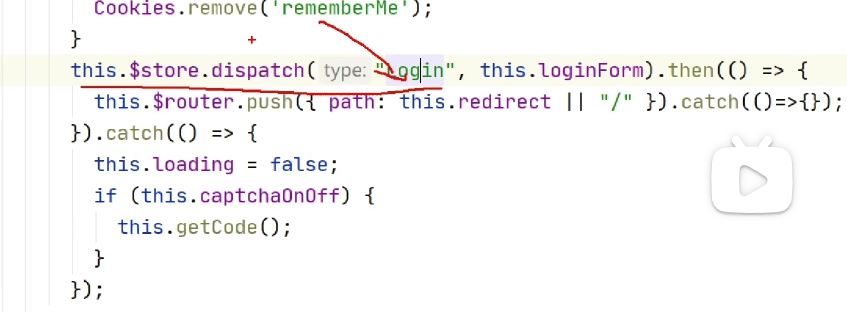
根据当前用户的权限获取动态路由



寻找对应出口：



首页数据加载



封装内容如上

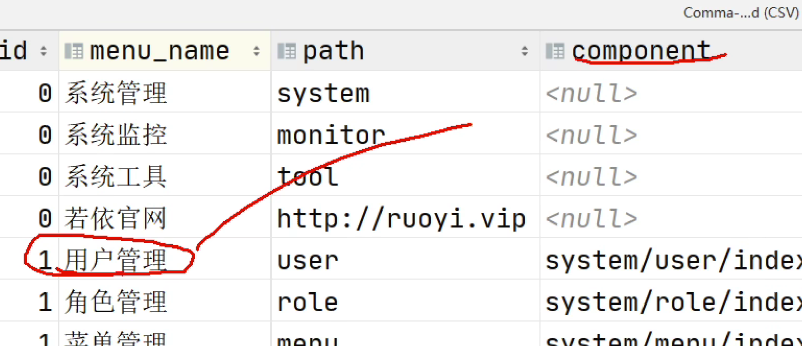
测边栏sidebarItem的具体代码：



首页内容：



数据库管理信息列表：



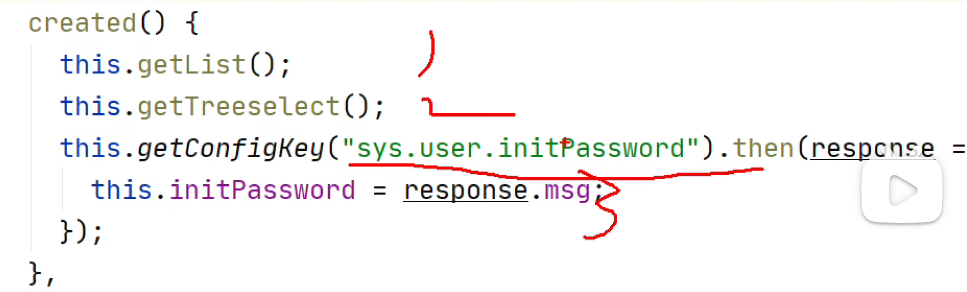
页面list和tree加载过程：



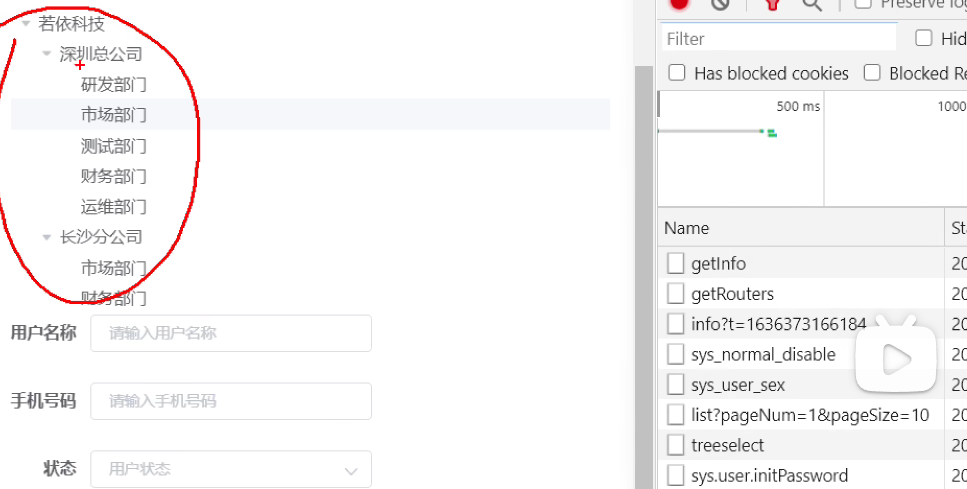
用户数据PageHelper分页

用户管理流程：

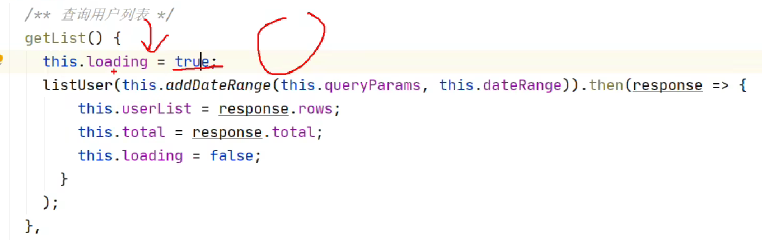
加载Vue页面-请求后台数据



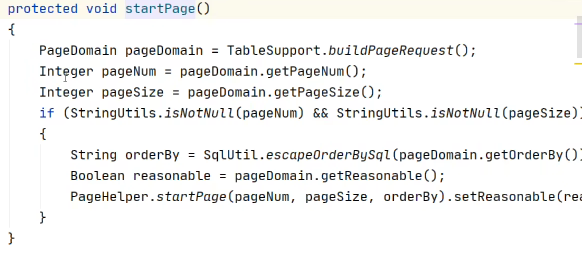
页面递归：



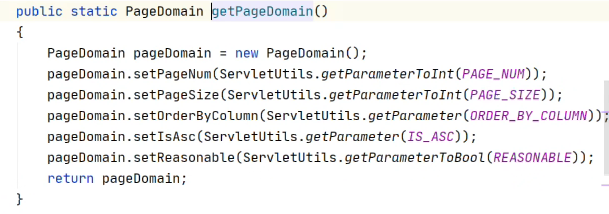
查询用户列表Getlist的方法:



Startpage分页方法：

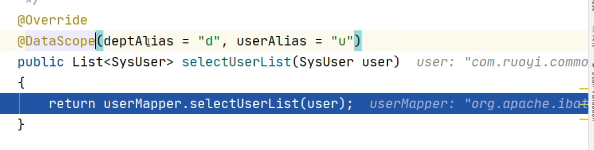


封装分页对象（注意相关嵌套）：



用户数据查询：

根据条件分页查询用户列表



Userservice.selectUserList(user)

注释：@Datascope(depAlias = “d”, userAlias = “u”)

给表设置别名的：sys\_dept d, sys\_user u

1. treeselect

流程：查出所有的部门数据->组装成树状结构

buildDeptTreeSelect:将10条记录组装成一个树状图

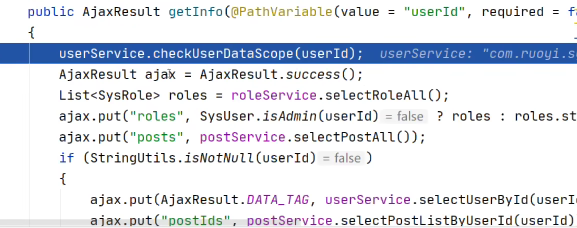




reset是表单重置

GetTreeselect获取部门树状图

getUser获取角色和部门信息



/添加用户的实现

/修改用户的实现

/删除用户的实现

/异步任务管理器

/代码自动生成