# Vue-cli3.0 + axios+mockjs实战入门初体验

实战项目开发中，我们经常采用前后端分离的开发模式，那么在没有后端接口时，我们是否可以独立于后端进行开发呢？答案是肯定的。mock便可以给你提供这样的体验，模拟了各种开发场景，提供了丰富的数据类型，支持生成随机的文本、数字、布尔值、日期、邮箱、链接、图片、颜色，让你如行云流水般舒适的摆脱后端进行开发。用法也是相当的简单，下面我们就在vue框架中体验一把mock的用法吧，飞一般的感觉，如梦初醒～

1. 下载node安装包并安装：https://nodejs.org/en/download/

安装vue脚手架：npm install -g @vue/cli

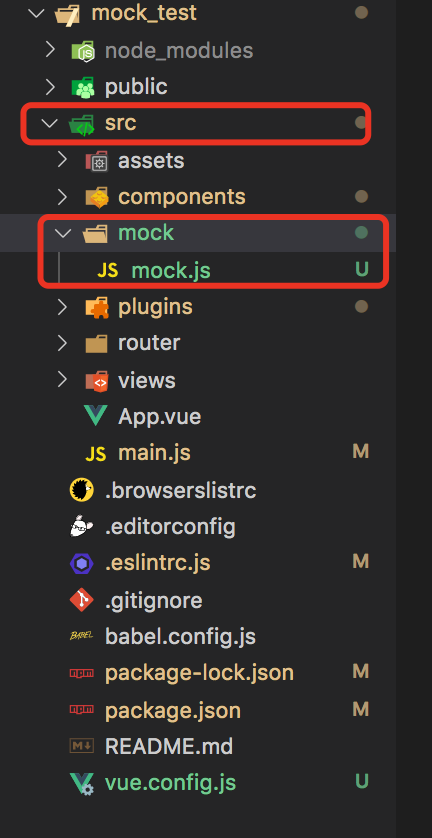
1. 使用vue-cli脚手架创建项目

vue ui

1. 安装依赖：mockjs



1. 在项目根目录中找到src目录，并在该目录下创建mock文件夹，同时在mock文件夹中创建mock.js文件，结构如图：



1. 打开mock.js文件，写入如下代码并保存：
2. // 引入mockjs
3. **import** Mock from 'mockjs'
4. // 使用mockjs模拟数据
5. Mock.mock('/api/data', (req, res) => {
6. **return** {
7. data: [
8. {
9. id: 1,
10. username: 'aaa',
11. password: 'aaa'
12. },
13. {
14. id: 2,
15. username: 'bbb',
16. password: 'bbb'
17. },
18. {
19. id: 3,
20. username: 'ccc',
21. password: 'ccc'
22. }
23. ]
24. }
25. })
26. 打开main.js引入mock.js文件

require('./mock/mock')

1. 安装axios依赖

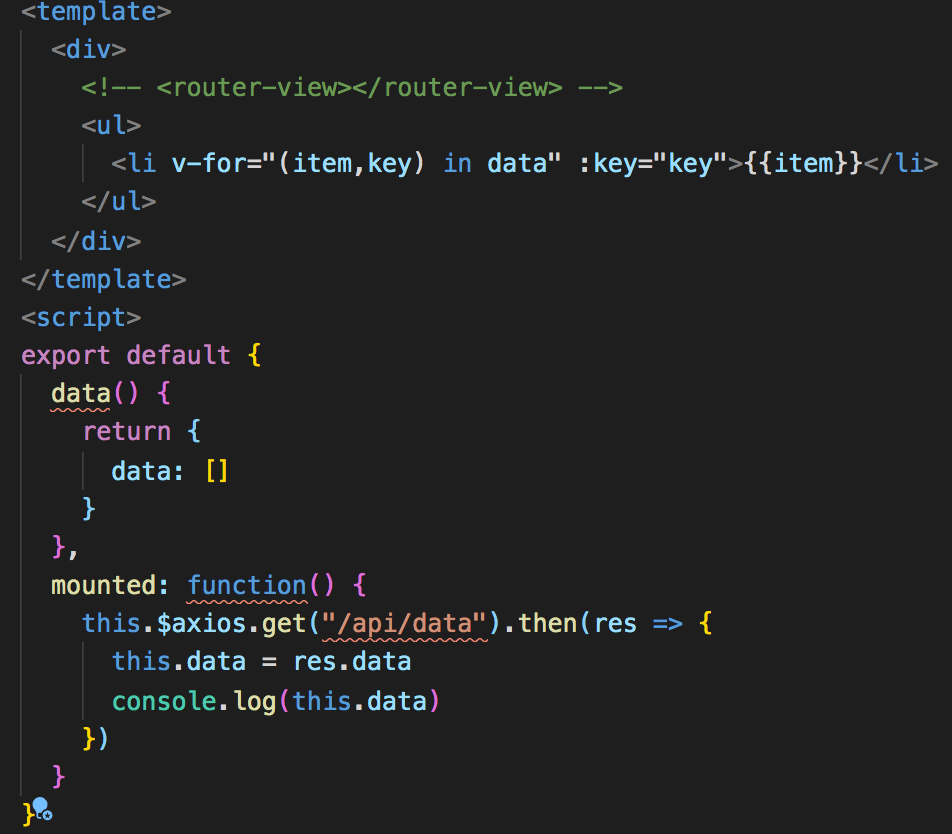


1. 打开main.js文件并引入axios，添加代码如下：

import axios from 'axios'

Vue.prototype.$axios = axios

1. 打开src/components中的HelloWord.vue组件，写入代码如下：



1. 通过脚手架启动项目，便可看到如下效果：



使用Vue-cli3.0 + axios+mockjs模拟后端请求就是这么简单，是不是很有成就感呢？Mockjs，so easy～