1. 在需要安装的文件夹下shift+右键，选择在此处打开命令窗口
2. 安装cnpm
3. npm install -g cnpm --registry=https://registry.npm.taobao.org
4. cnpm install --global vue-cli

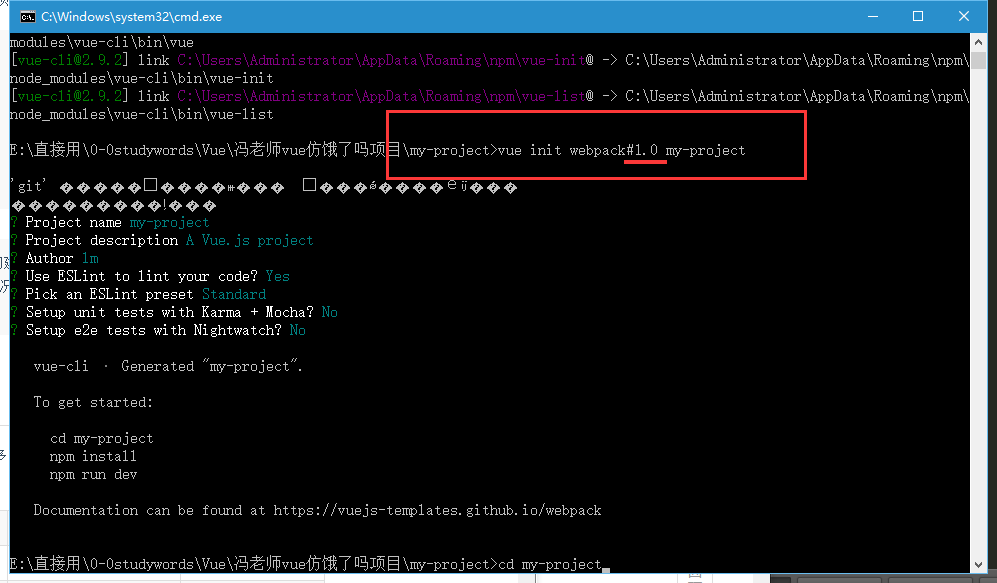
npm install module\_name -S    即    npm install module\_name --save    写入dependencies

npm install module\_name -D    即    npm install module\_name --save-dev 写入devDependencies

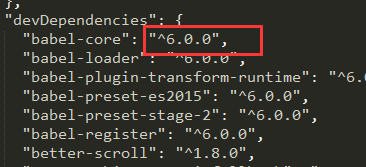
npm install module\_name -g 全局安装(命令行使用)

1. vue init webpack my-project

设置安装版本1.0版本

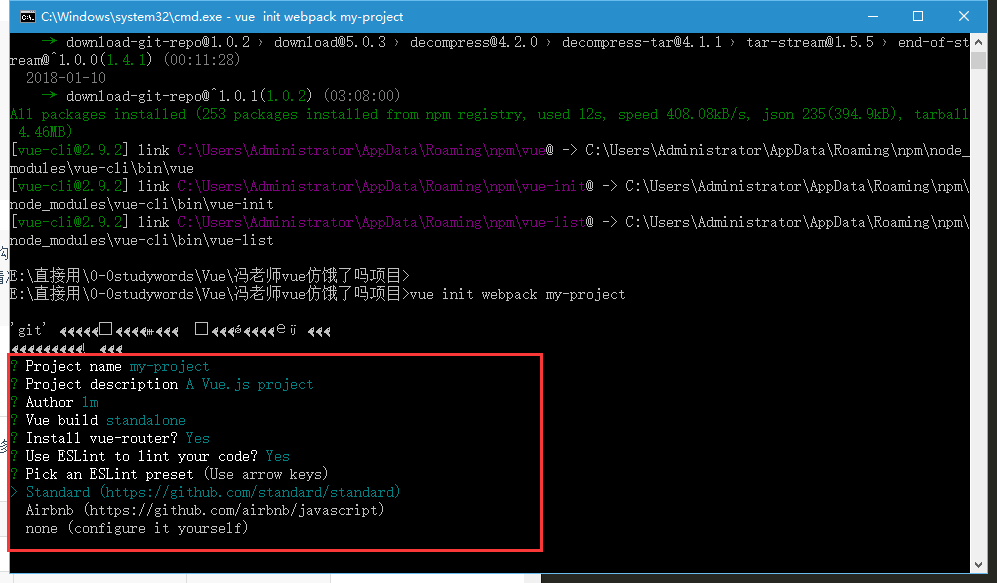


Webpack#1.0

备注：安装后想更换版本，打开package.json修改参数

修改后cnpm install安装一下在运行即可

1. 回答问题



NO 不验证代码格式

1. 如何安装To get started:

就是下面的四步图2，第三部可以省略图1；因为之前安装了cnpm,所以用cnpm代替npm，见图3



图1

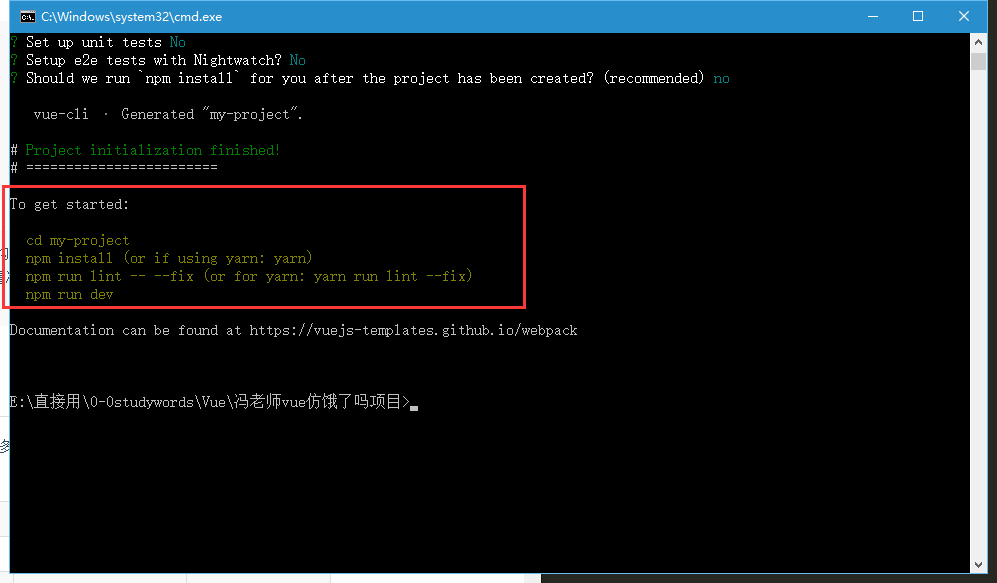


图2

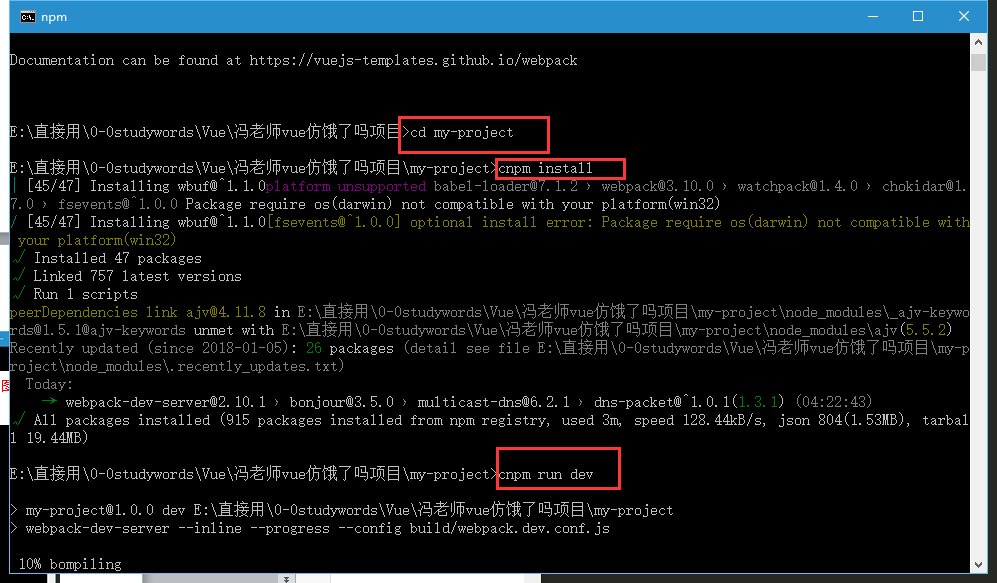
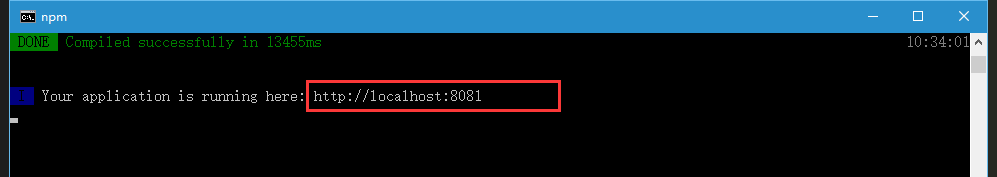
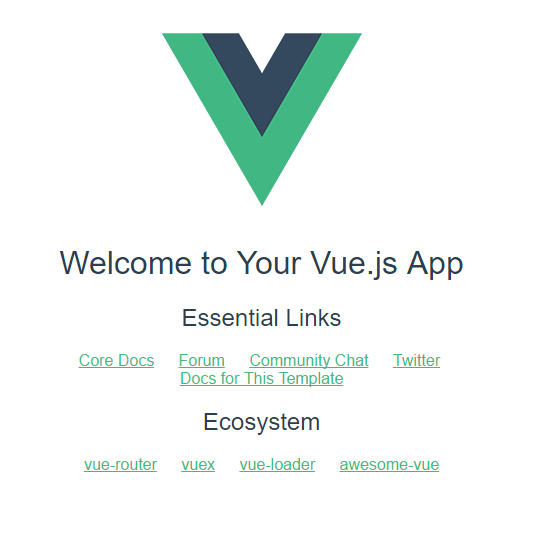


图3

1. 安装完毕显示，在浏览器打开这个地址，显示vue的 Demo页面





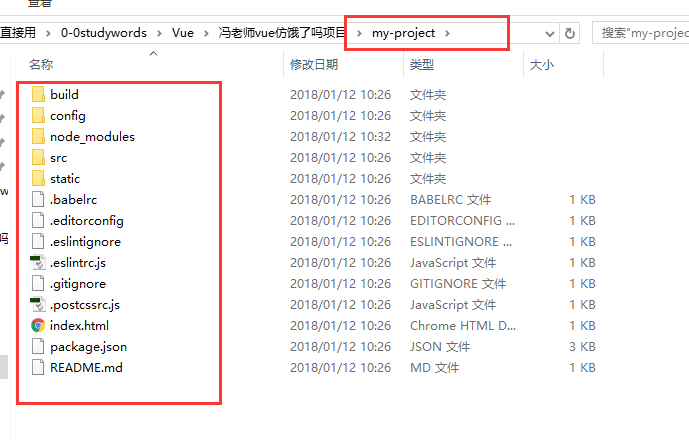
8-2、别人的vue项目怎么运行，

复制文件后

cnpm install（原理是：根据他的package.json配置，安装依赖）

cnpm run dev 即可

1. 文件夹下安装了这些文件



**项目知识点：**

1、Cnpm安装sass

② 安装sass 依赖包 ，在cmd界面输入：

npm  install sass-loader --save-dev

npm install node-sass --sava-dev

③ 在build文件夹下的webpack.base.conf.js的rules里面添加配置

{

test: /\.scss$/,

loaders: ['style', 'css', 'sass']

}

④ 使用scss时候在所在的style样式标签上添加lang=”scss”即可应用对应的语法，否则报错

⑤ scss使用测试：如下测试修改字体颜色

<style lang="scss">

$color:red;

div {color:$color;}

</style>

2、cnpm安装resource（请求用）

cnpm install vue-resource --save-dev

3、cnpm安装router（路由器）0.7.13是版本

cnpm install vue-router@0.7.13 --save-dev

4、卸载插件



cnpm uninstall 插件名称

5、cnpm 安装better-scroll滚动插件

cnpm install better-scroll --save-dev

使用方法说明：

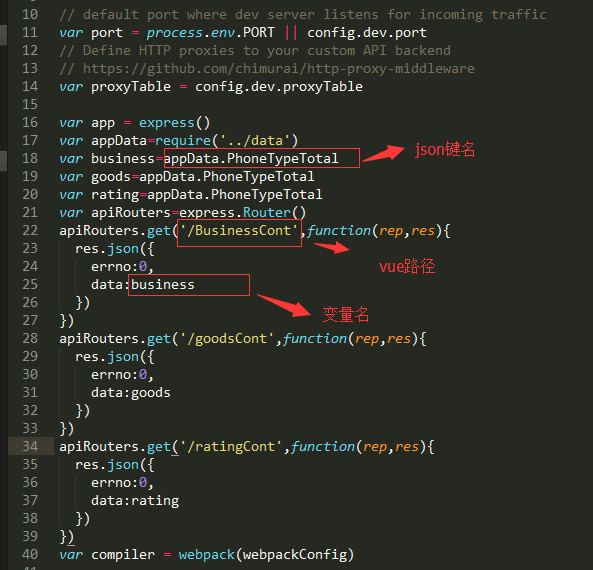
1. 安装、引用
2. 给要添加滚动的父元素加上v-el: menu-wrapper

备注：menu-wrapper固定格式，小写用‘-’连接

1. 写一个初始化方法，调用一下，初始化的时候传递第一个参数是this.$els.menuWrapper（驼峰格式），这时候就能实现滚动的效果，但是如果是左右两侧滚动，左侧菜单，右侧内容，滚动右侧左侧对应菜单名称高亮显示，点击左侧菜单右侧滚动显示对应内容还要继续设置







6、cnpm安装vuex插件

cnpm install vuex –save vuex

（使用的时候和router相同也是import引入、use、实例化、放到全局实例化对象）

7、本地模拟向后台请求数据

方法一：这种方法只能get。Post报错

代码如下：

**const** express = **require**('express')

**const** app = express()

**var** appData = **require**('../mock/goods.json')//加载本地数据文件

**var** apiRoutes = express.Router()

app.**use**('/api', apiRoutes)

//然后找到devServer,在里面添加

before(app) {

app.**get**('/api/appData', (req, res) => {

res.json({

errno: 0,

data: appData

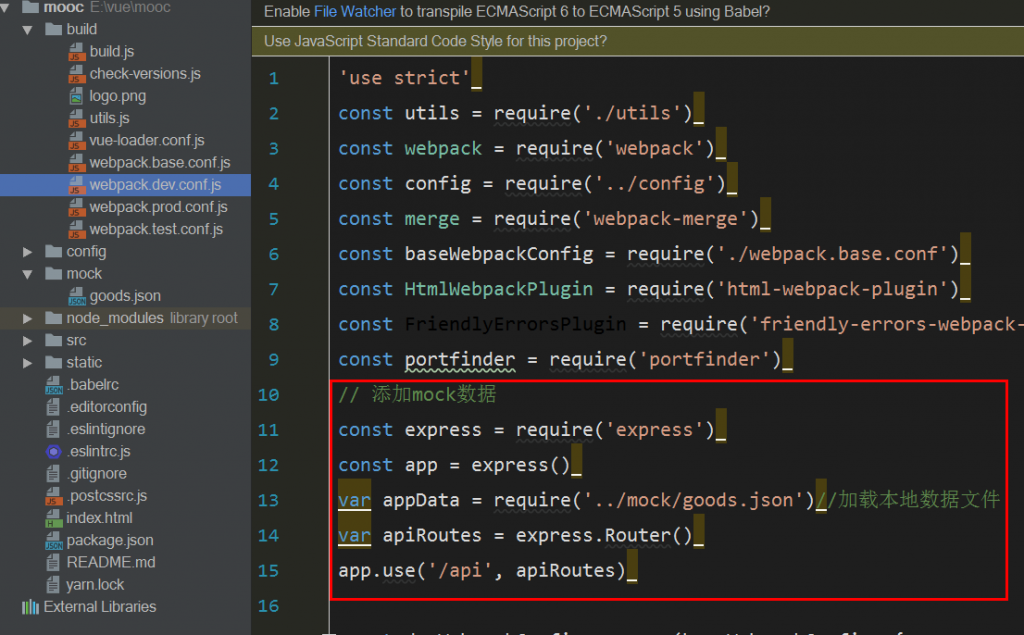
})//接口返回json数据，上面配置的数据appData就赋值给data请求后调用

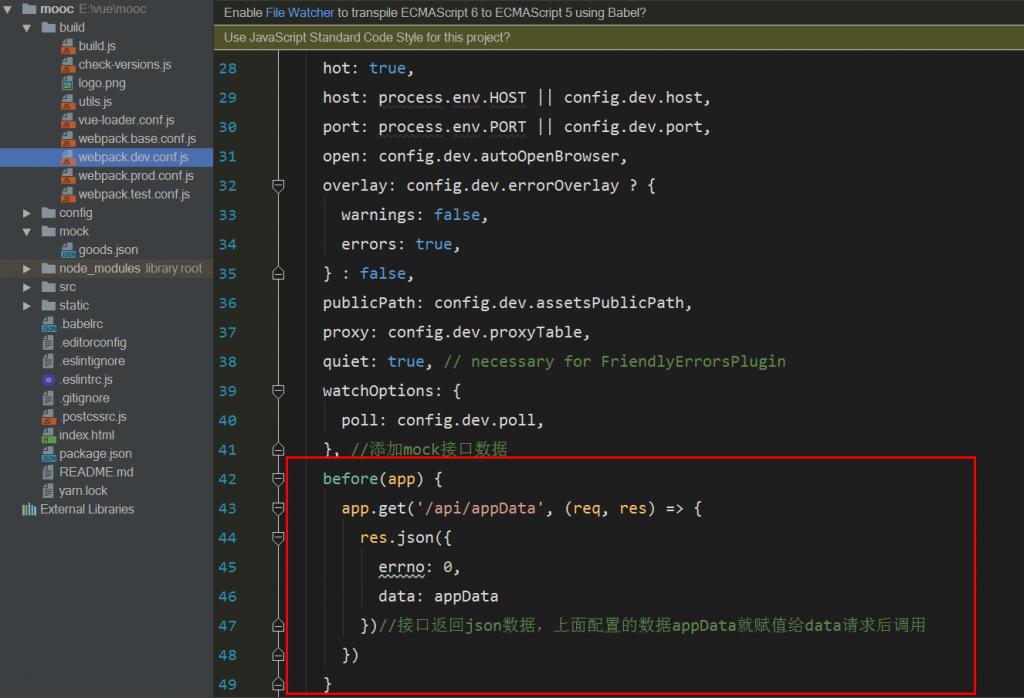
})

}

然后npm run dev，一定要重启 一下就可以http://localhost:8080/api/appData 访问了

截图如下：





方法二：

视频：6章4-7的60%部分

https://blog.csdn.net/zhouweixue\_vivi/article/details/74783934

**8、cnpm run build打包的时候，index.html引入文件没有引号，和路径问题**

①、没有引号：

<link href=/css/xxx.css rel=stylesheet>

解决办法：

找到webpack.prod.conf.js，在webpack.prod.conf.js找到minify

把minify中的  removeAttributeQuotes: true改为

 removeAttributeQuotes: false（如果该方法没有用那就把整个minify去掉再试一下）

然后在执行打包命令：npm run build

执行完打包后index.html中的css和js引入时就会有引号

<link href="/css/xxx.css" rel="stylesheet">

②、路径缺 ” . ”

<link href="/css/xxx.css" rel="stylesheet">应该是<link href="./css/xxx.css" rel="stylesheet">



重新cnpm run build之后正常

1. axios请求的时候需要注意的地方：

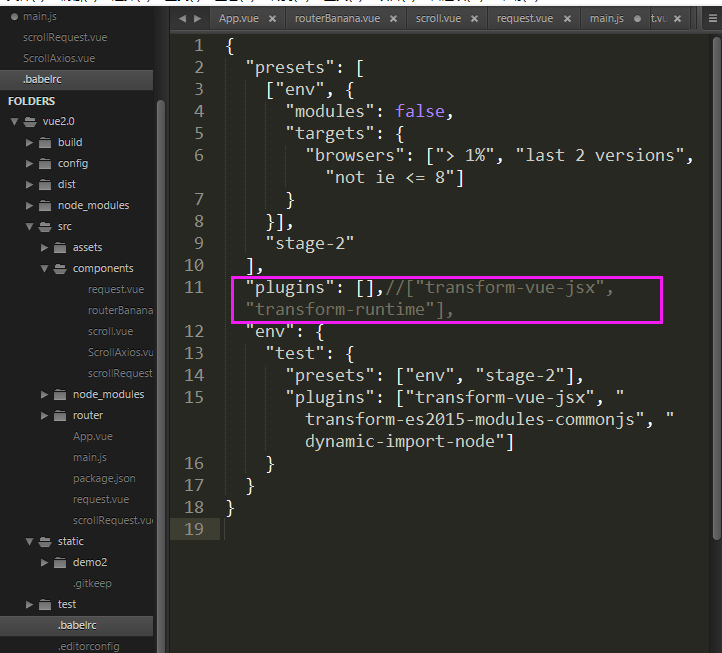
第一步：cnpm install axios

第二步：

import axios from 'axios'

Vue.prototype.$ajax =axios

(如果引用后报错：把babelrc.json文件中的plugins的数组清空)

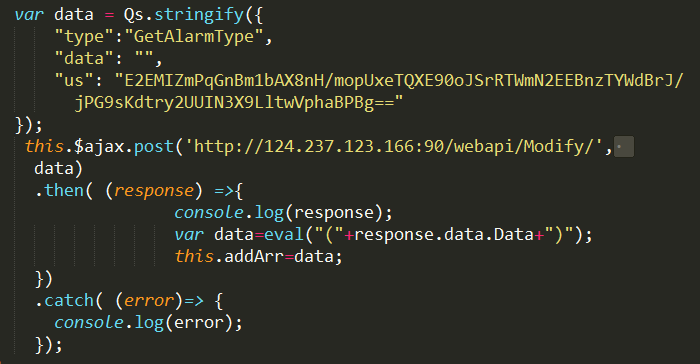


第三步：

在需要请求的vue中引入Qs(axios安装时自带，直接引入即可)

import Qs from 'qs'





Var data=Qs.stringify({

这里面是要传的参数

})

this.ajax.post({‘url’, data})

.then((res)=>{成功回调})

.catch((err)=>{失败回调})