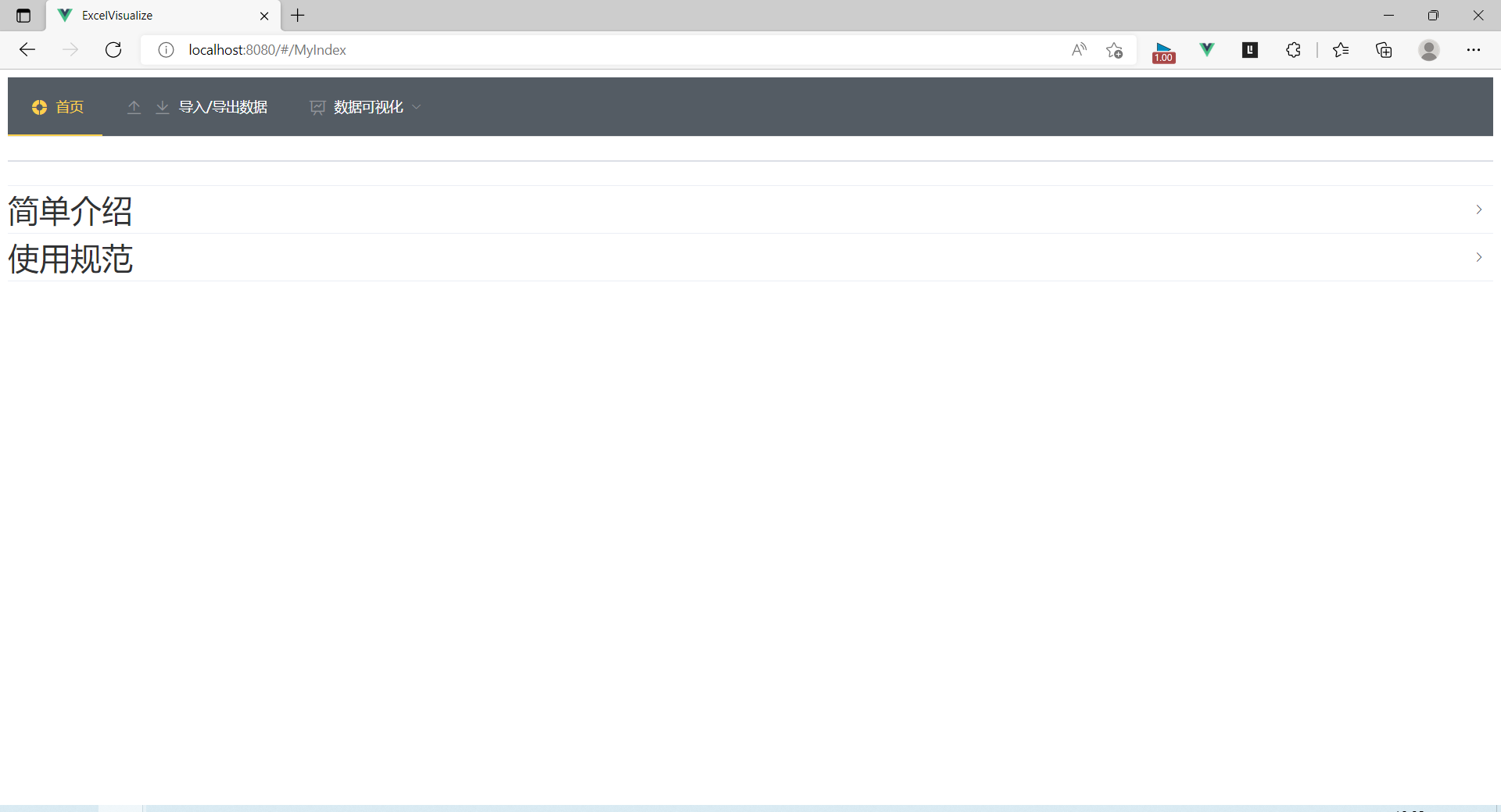
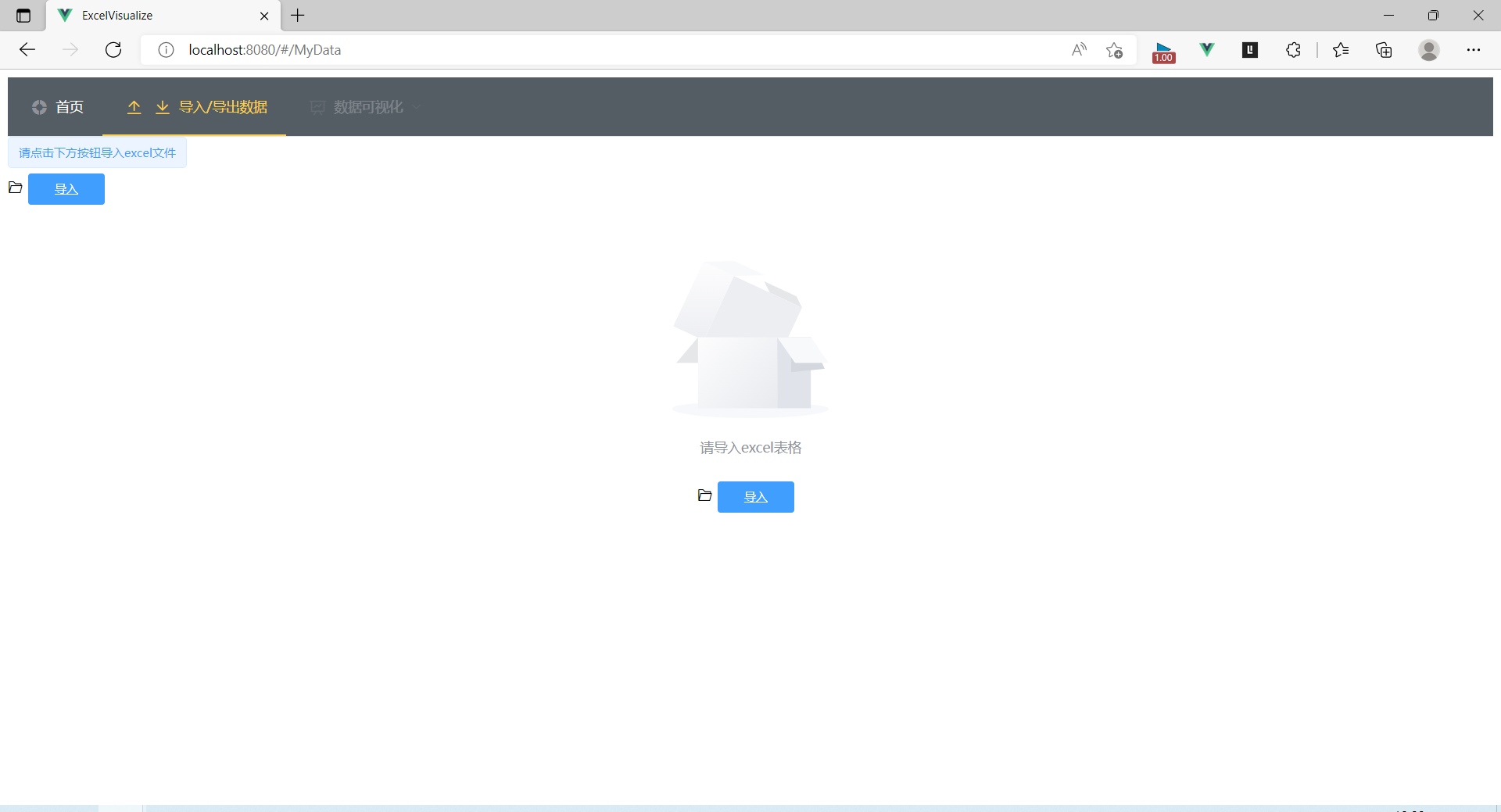
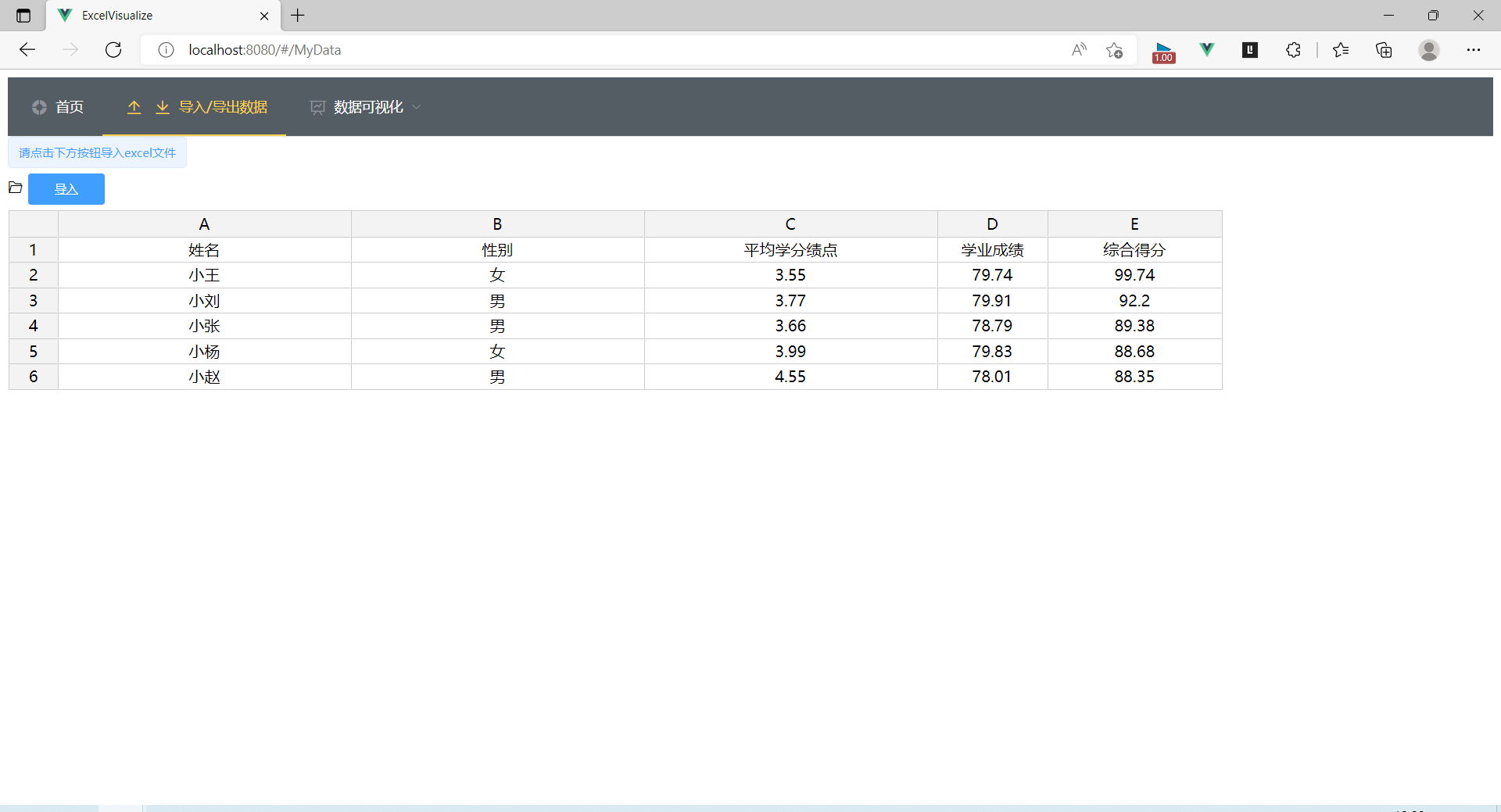
# 项目总体简介

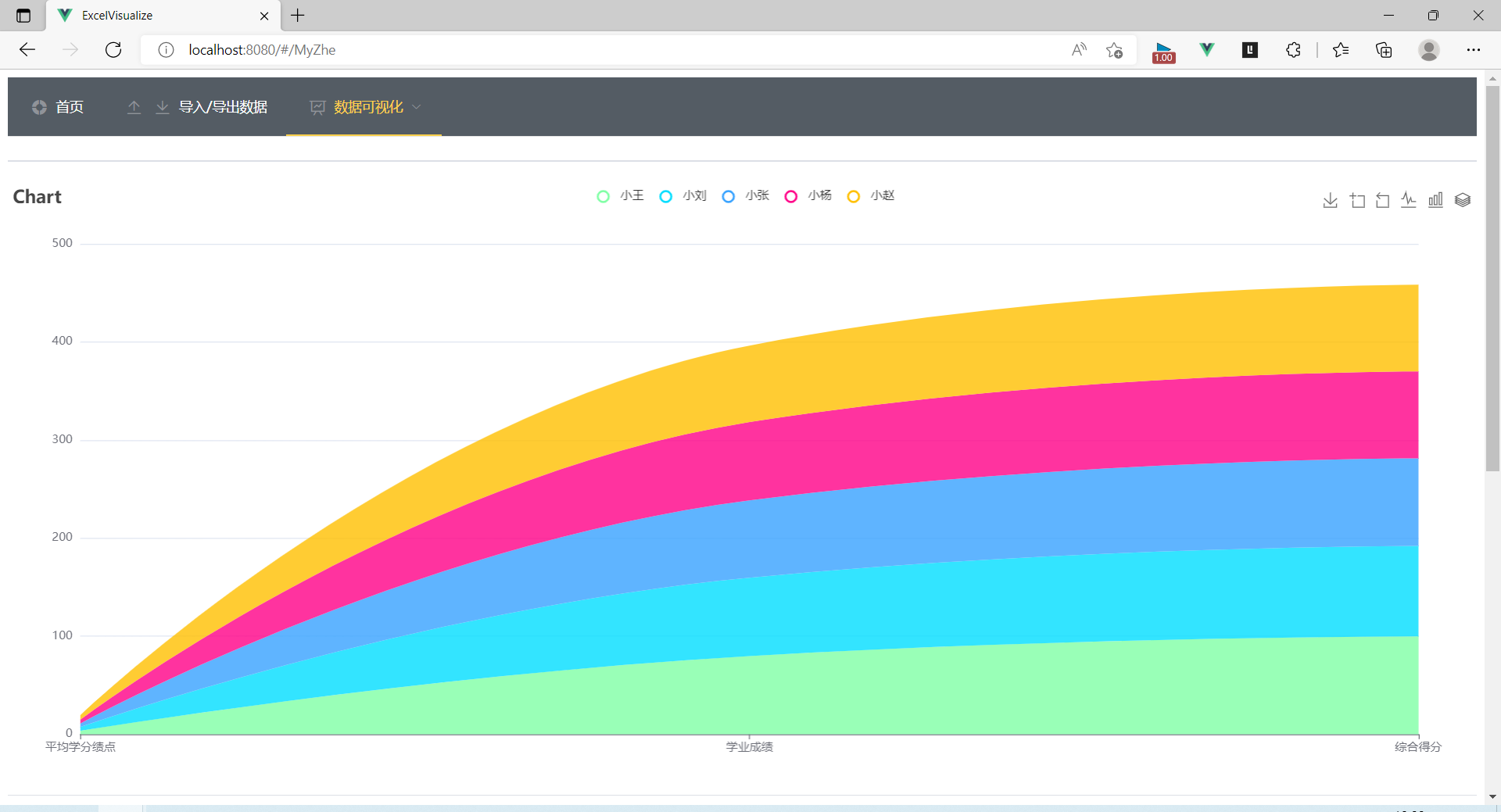
设计该项目是觉得可以更好的帮助用户将**自己**平时使用的excel表格中的数据进行可视化，可以**服务于广大用户**，而**不是**做仅仅针对某一数据源的**现成的固定的数据**进行可视化，仅需鼠标动一动便可以导入自己的excel表格数据，将其制作成图表，保存成图片，还可以选择多种不同的形态，折线图柱状图饼图，堆叠型和非堆叠型，***总有一种是适合用户数据的***。

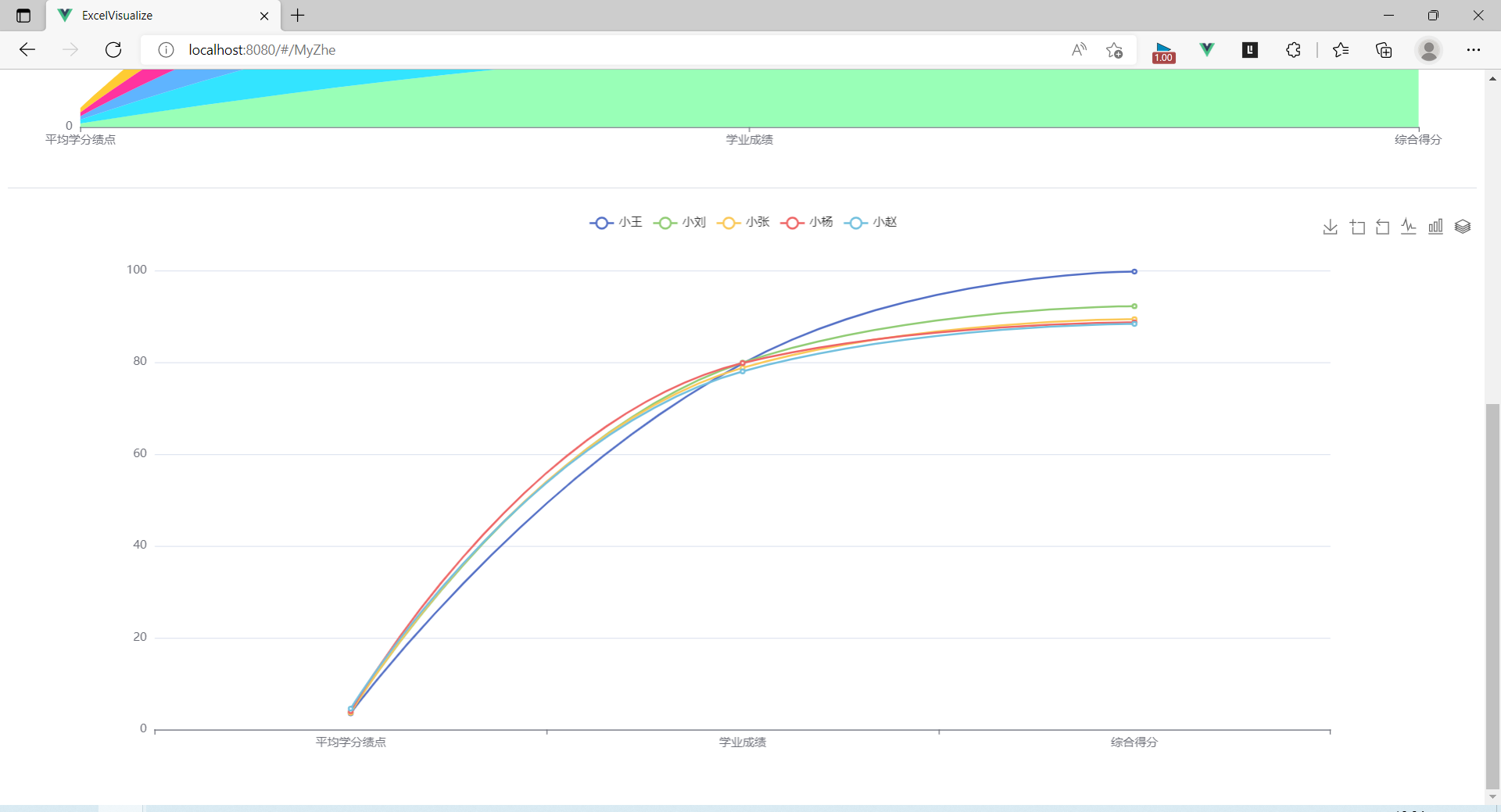
具体一些的使用介绍在项目的首页有写（见下图）

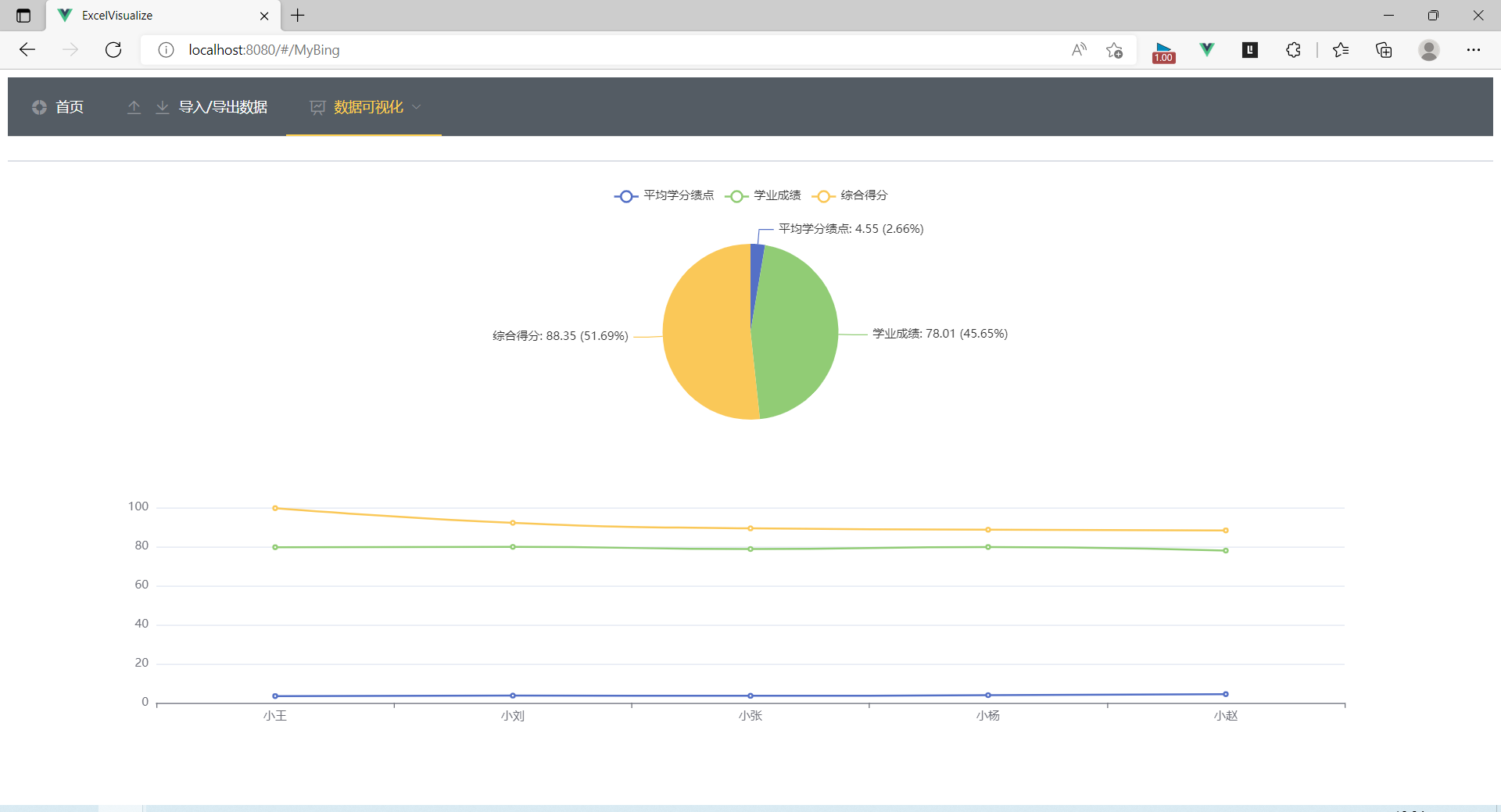






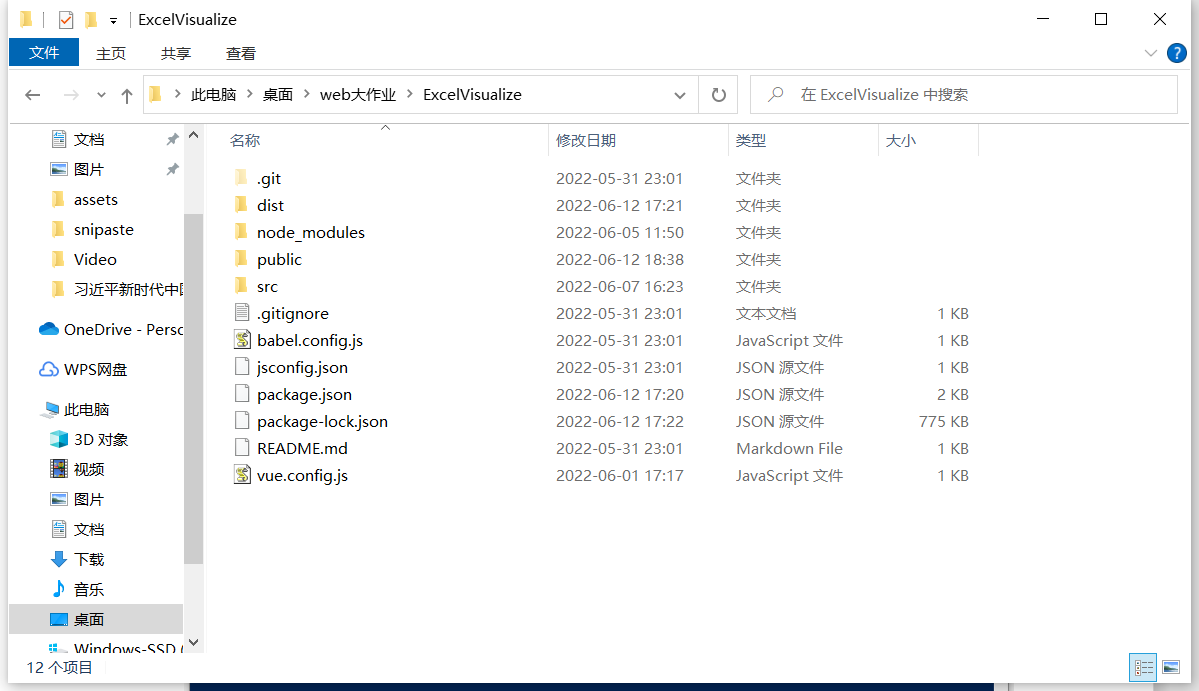






该项目采用了Vue.js、Echarts.js、Element UI、jQuery、Jexcel(网页展示excel表格)、XLSX(读取excel文件)等技术，实现了让用户在网页端**导入自己的Excel文件**，对文件修改后另存为本地文件，也可以**将Excel文件中的数据进行可视化处理**，生成图表。

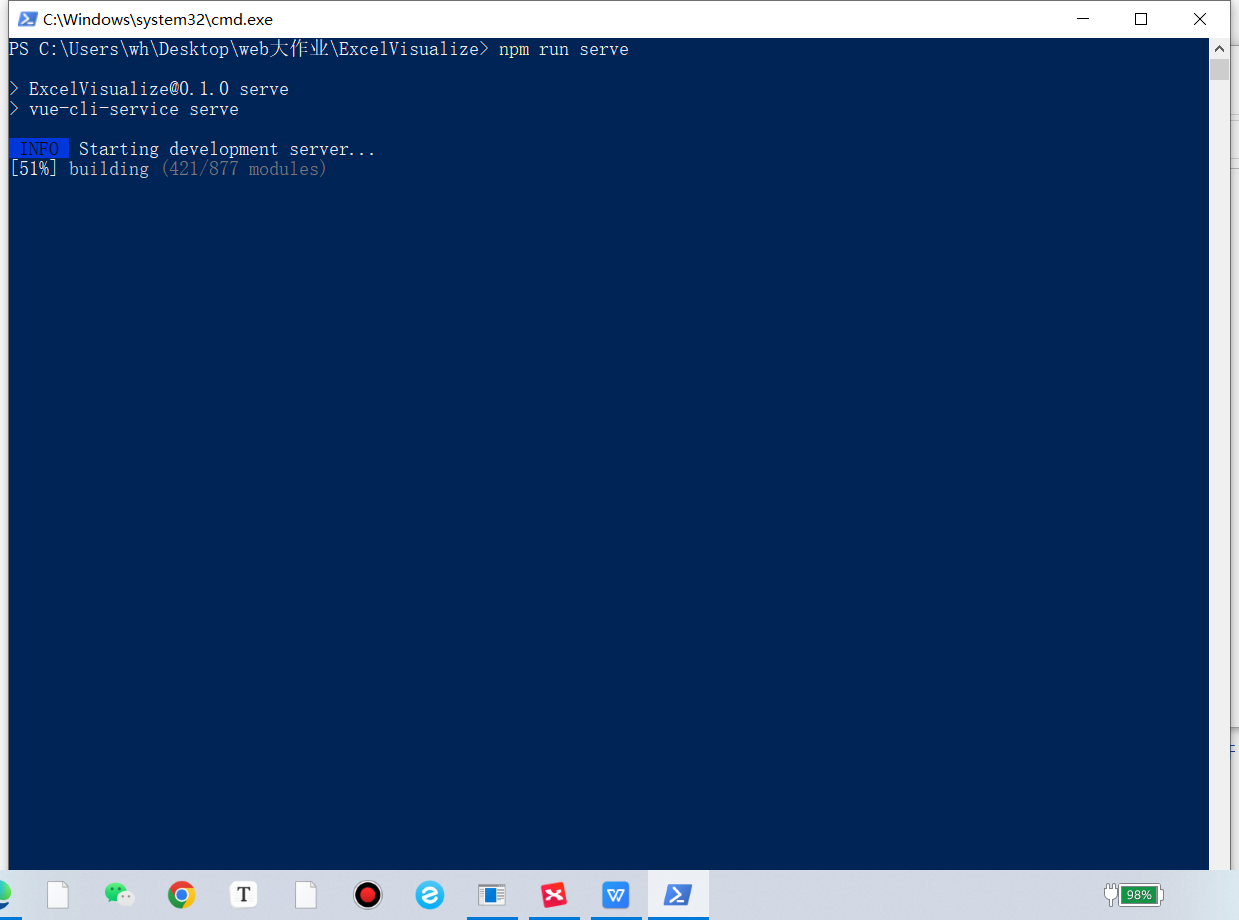
运行项目

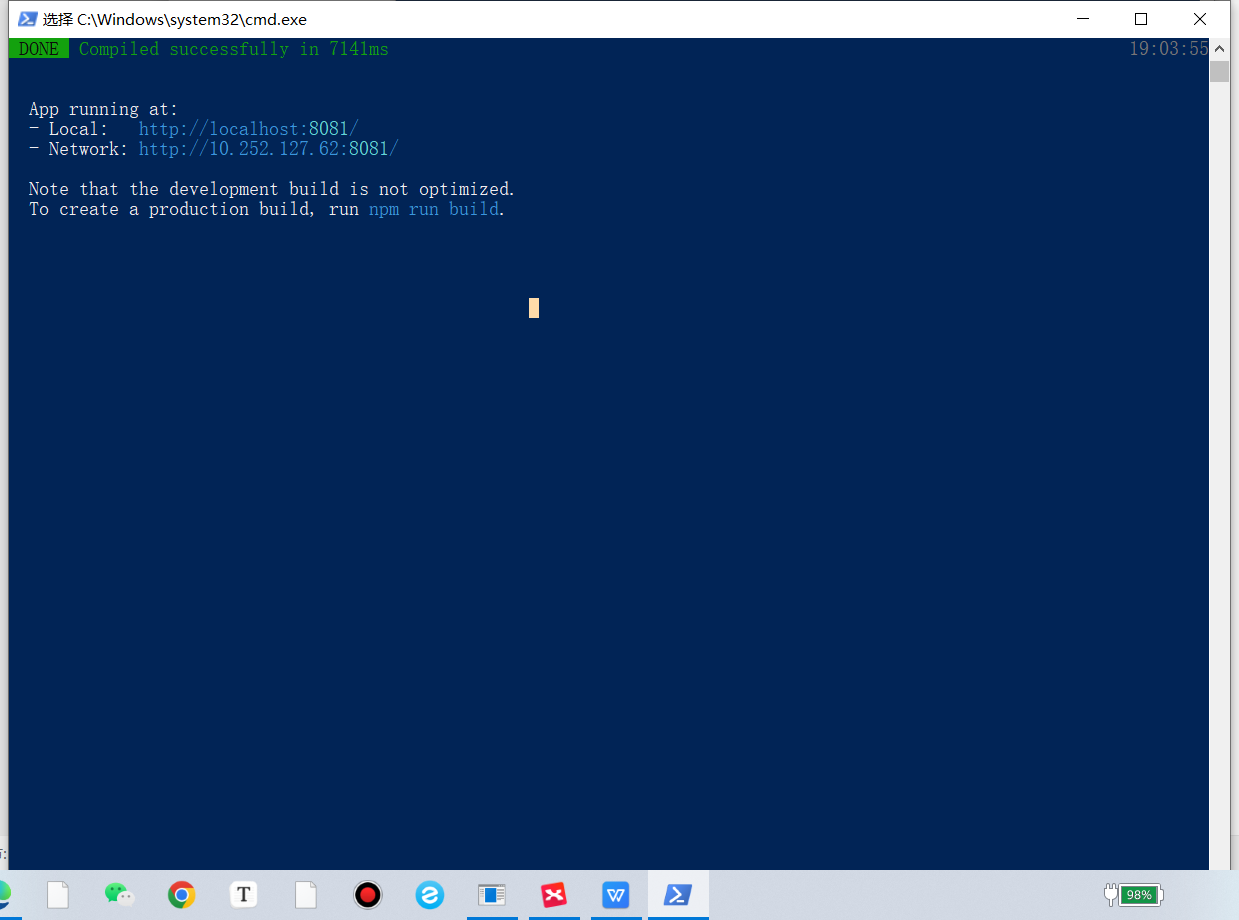


在该页面按住shift+单击鼠标右键，选择 “在此处打开Powershell窗口”，先输入npm i

即可自动安装项目所需依赖包(即生成node\_modules文件夹)

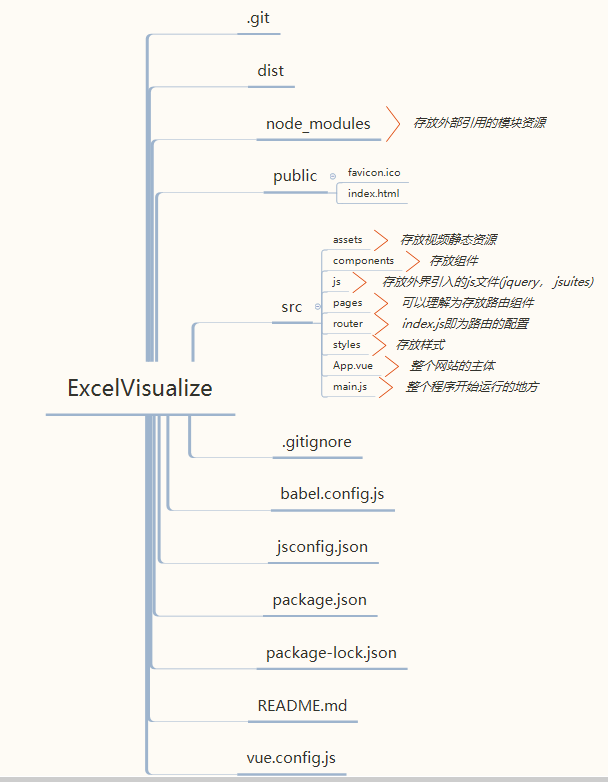
在命令行输入npm run serve





如图所示即为成功运行项目了，通过Local对应的网址即可在浏览器访问

# 2.项目文件展示



# 3.项目技术选择

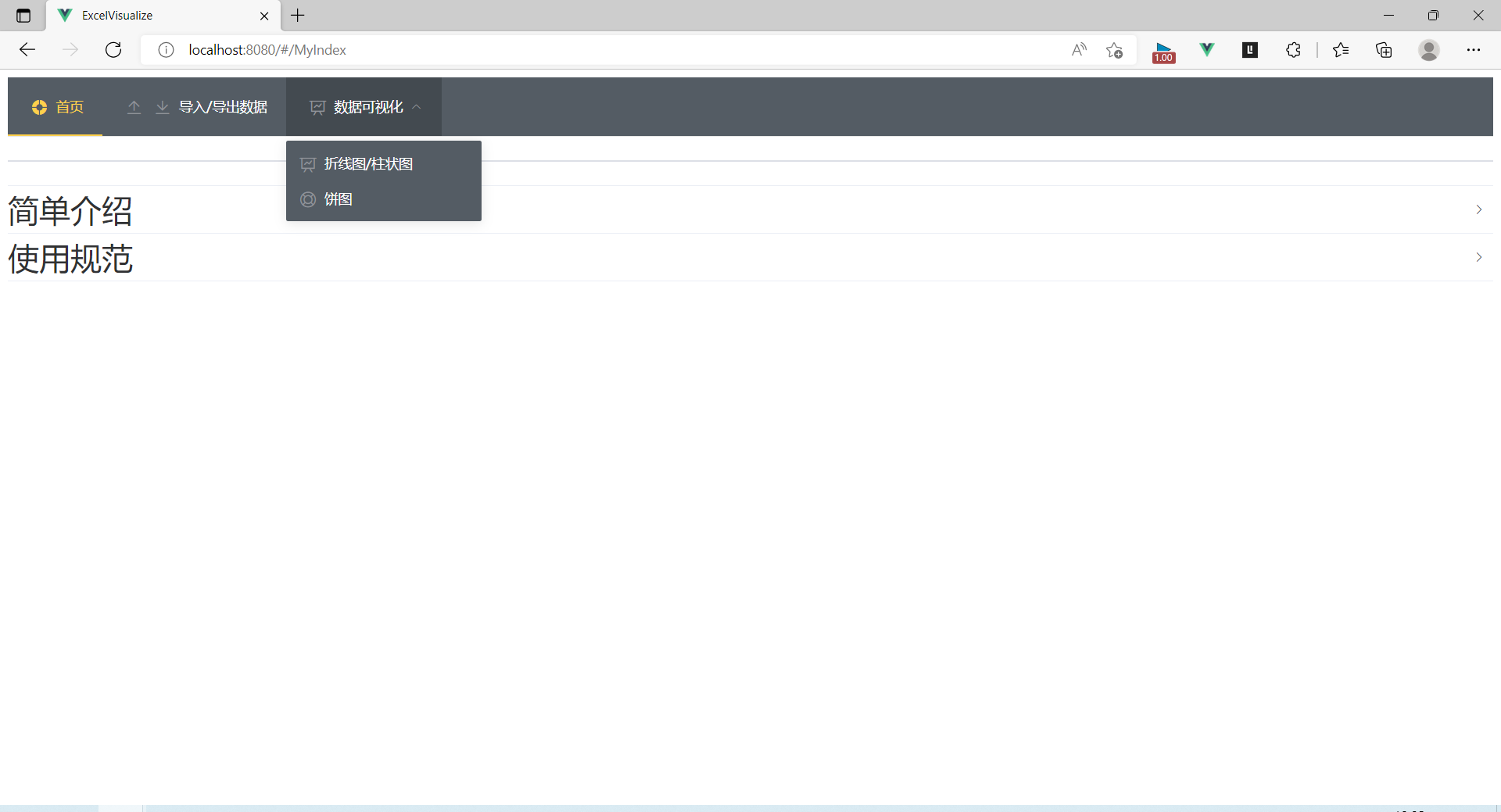
## 3.1 Vue.js

##### Vue框架主要具有几个优点

**能够提高开发效率**

**能够使整个项目更加模块化，多个内容一体化，方便管理和阅读**

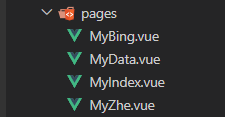
这是App.vue的部分代码，可以看到在Vue框架下，App.vue文件很好的对把不同部分联系了起来



**实现路由操作(vueRouter)，比用好几个html文件之间来回跳转更加直观，符合逻辑**

这是在router/index.js中实现路由器的代码，主要有四个路由器组件，四个路由器组件有对应的.vue文件





还需要在main.js文件中注册自己写的路由器



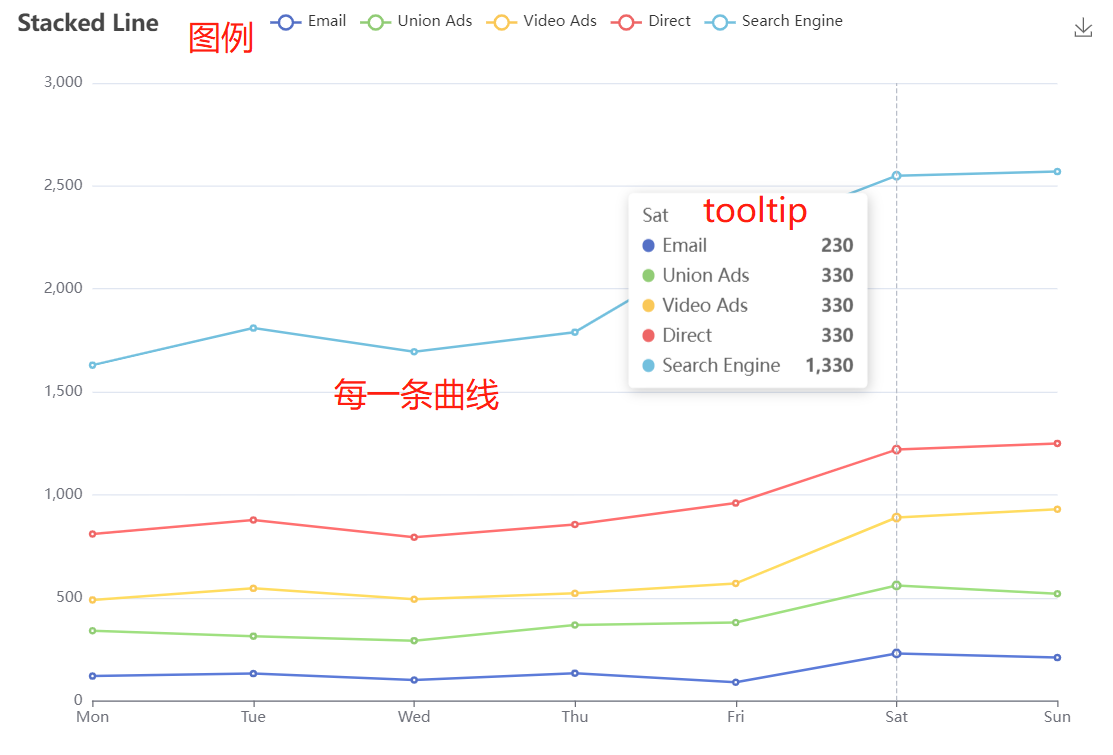
## 3.2 Echarts

**大致的用法**是这样的



下面是**对配置项的具体解读**





## 3.3 Element UI

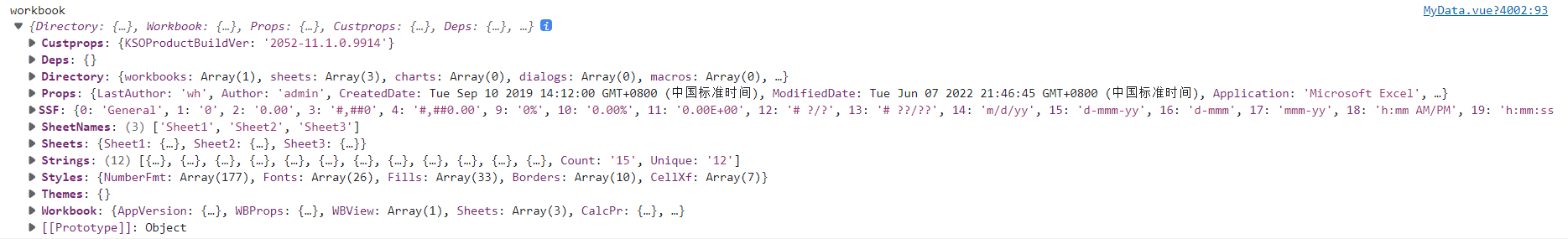
Element UI可以提供给开发者更简便美观的样式设计

可以实现用较少的代码便实现不错的效果

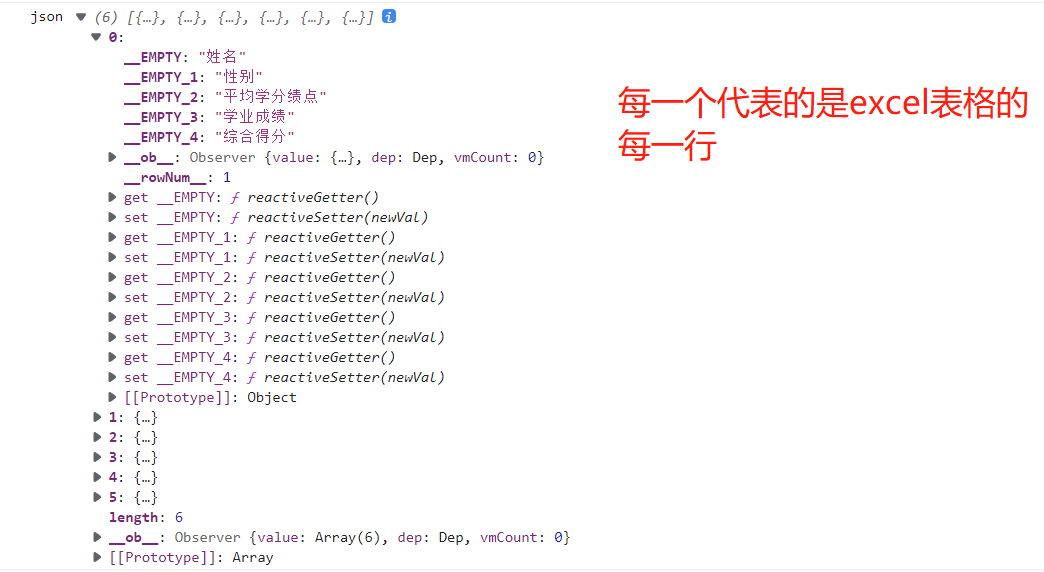


## 3.4 XLSX









## 3.5 Jexcel

用于将读取的excel表格的数据形成excel表

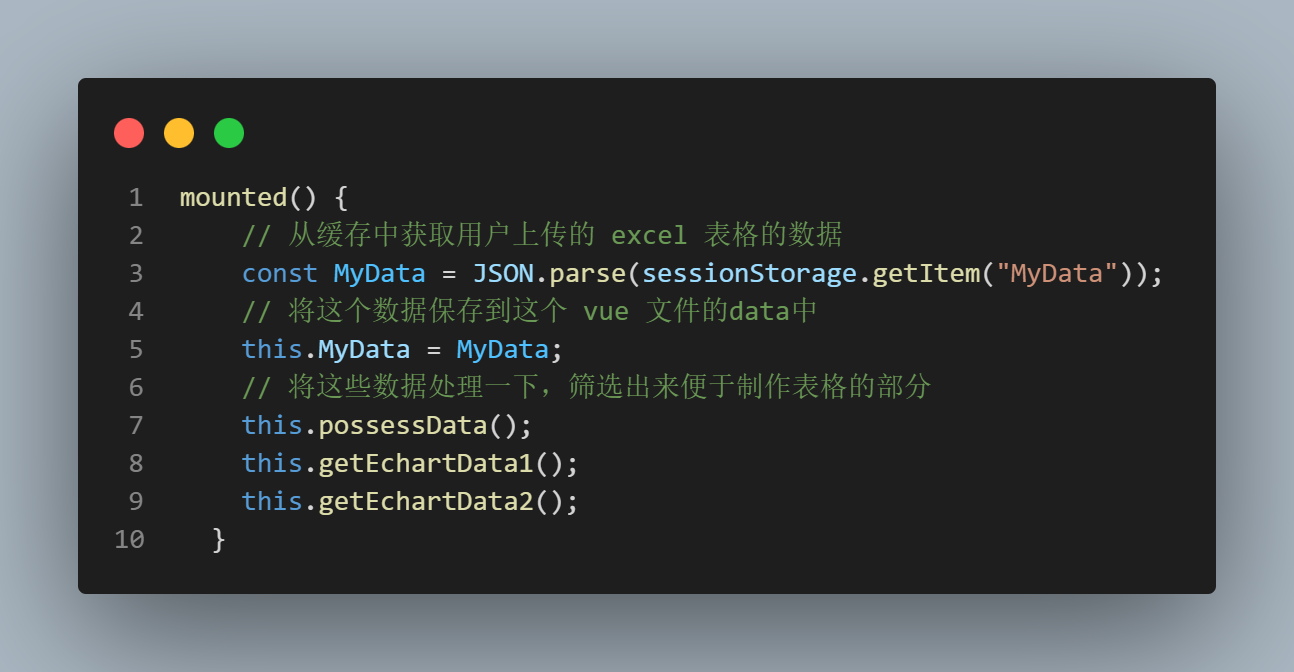


# 4.关键部分代码解读

读取用户上传的excel表格并用Jexcel生成web端的excel部分的代码在3.4 XLSX 和 3.5 Jexcel的部分中有阐述了

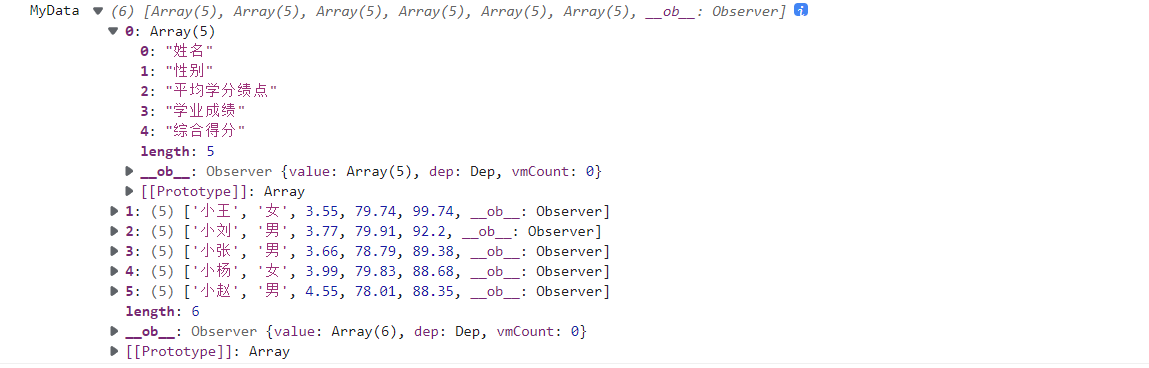
接下来具体讲讲如何利用用户上传的数据**动态生成配置项**并应用在表格上

### 4.1 对于折线图/柱状图部分



首先是要获取之前在 MyData 路由组件处理好存储在缓存中的数据，然后调用三个自己写的函数，分别是要**先处理这份数据使有用的数据留下**，并且能够**有一个比较好的数组形式**，方便后续利用。

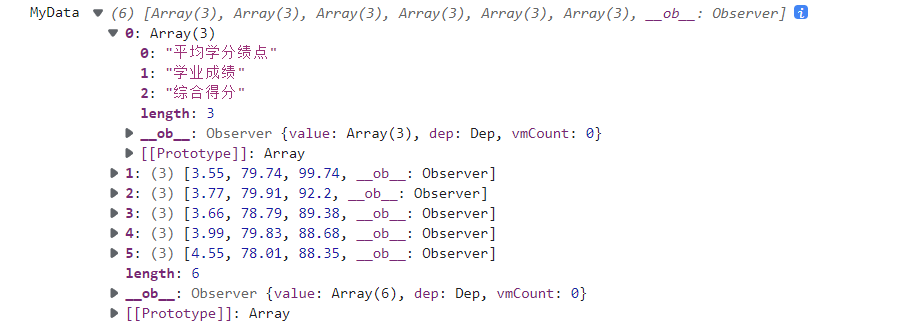
首先看一下原数据是怎样的

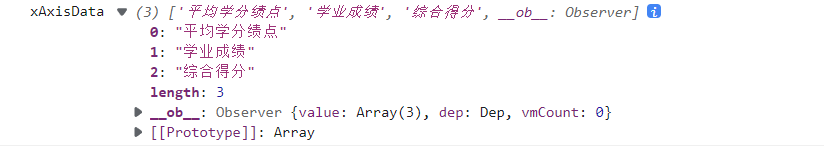


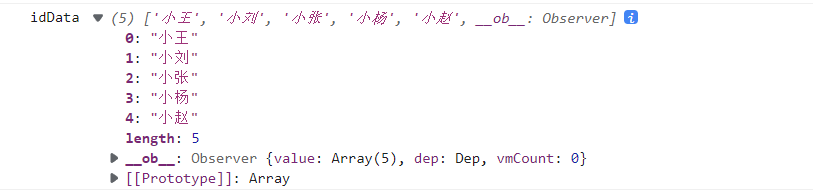
能够更好的明白应当如何处理数据



下面是处理后的数据的示例







**再来分析分析生成图表的函数getEchartsData中的细节**

**可以点击下方图片然后点击右侧放大镜来看，因为代码量挺大的只能这样大段截图**



### 4.2对于饼图部分

有大量的代码是重复的，在折线图/柱状图部分也有完整详细的分析，故在此不过多赘述





**在项目的首页还有许多的介绍，关于如何更好的使用这个网站**

**更多请移步项目**