**DataMonitor观测数据可视化管理系统  
开发文档**

**珠海市动立科技有限公司**

**2023年**

**目 录**

[**1** 系统概述 3](#_Toc131863683)

[1.1 系统功能 3](#_Toc131863684)

[1.2 开发技术 3](#_Toc131863685)

[**2** 配置开发环境 4](#_Toc131863686)

[2.1 安装Node开发环境 4](#_Toc131863687)

[2.2 安装VUE开发环境 5](#_Toc131863688)

[2.3 安装VSCode开发工具 5](#_Toc131863689)

[**3** 系统开发与调试 7](#_Toc131863690)

[3.1 创建项目 7](#_Toc131863691)

[3.2 开发代码 8](#_Toc131863692)

[3.3 运行程序 10](#_Toc131863693)

[**4** 系统功能开发 12](#_Toc131863694)

[4.1 配置vue.config.js参数 12](#_Toc131863695)

[4.2 添加Vue Router路由 13](#_Toc131863696)

[**5** 系统发布 15](#_Toc131863697)

# 系统概述

## 系统功能

DataMonitor观测数据可视化管理系统是一个基于百度地图库开发的WebGIS系统，采用了Node.js+VUE3开发技术框架，实现了基于WebGIS的观测数据监测业务的可视化展示及操作管理功能。

## 开发技术

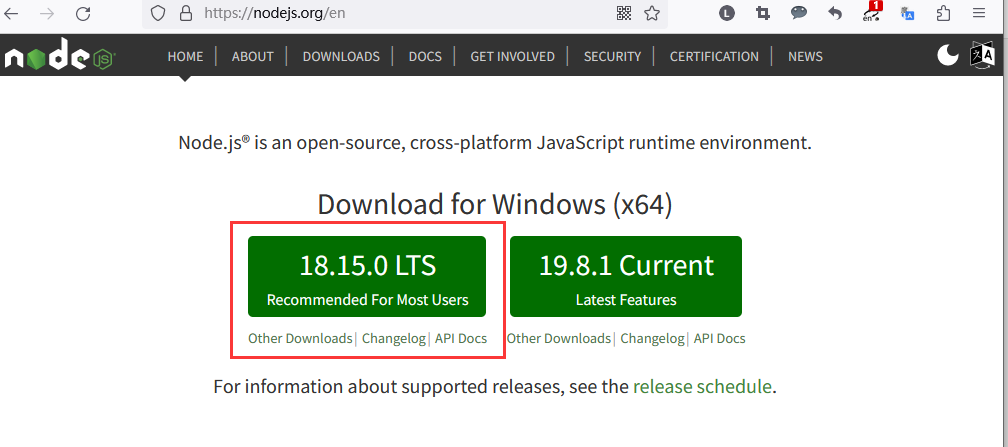
本系统开发的主要技术栈如下表所示。

|  |  |
| --- | --- |
| **运行环境** | |
| Node.js | v16.13.1 |
| Vue3 | 3.2 |
| **开发工具** | |
| IDE | VSCode |
| 浏览器 | FireFox |
| **Vue插件** | |
| "echarts": "^5.4.2",  "echarts-liquidfill": "^3.1.0",  "jquery": "^3.6.4", | |

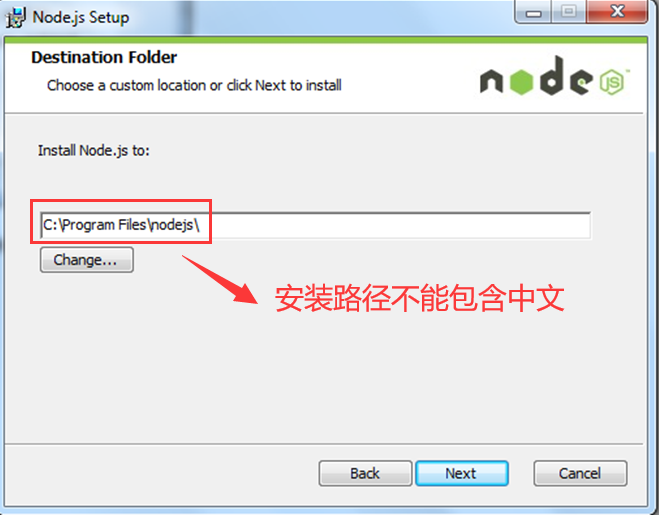
# 配置开发环境

## 安装Node开发环境

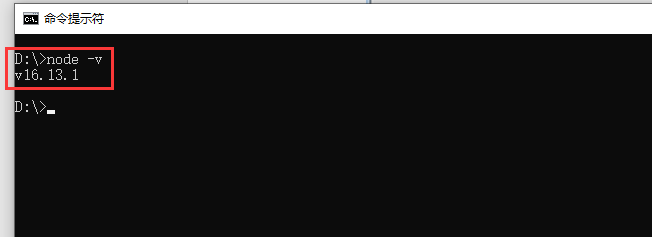
1、下载稳定版本的Node安装程序，官网地址为：<https://nodejs.org/en>。



2、下载Node安装程序后，按提示一步步完成安装即可，需要注意的是安装目录路径不能包含中文。



3、安装完成后，启动一个CMD命令行窗口，运行“node -v”命令查看安装信信息，如下图所示。

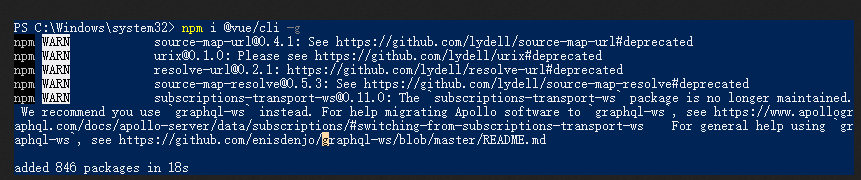


## 安装VUE开发环境

1、启动一个CMD命令行窗口，运行“npm install vue -g”开始安装vue。

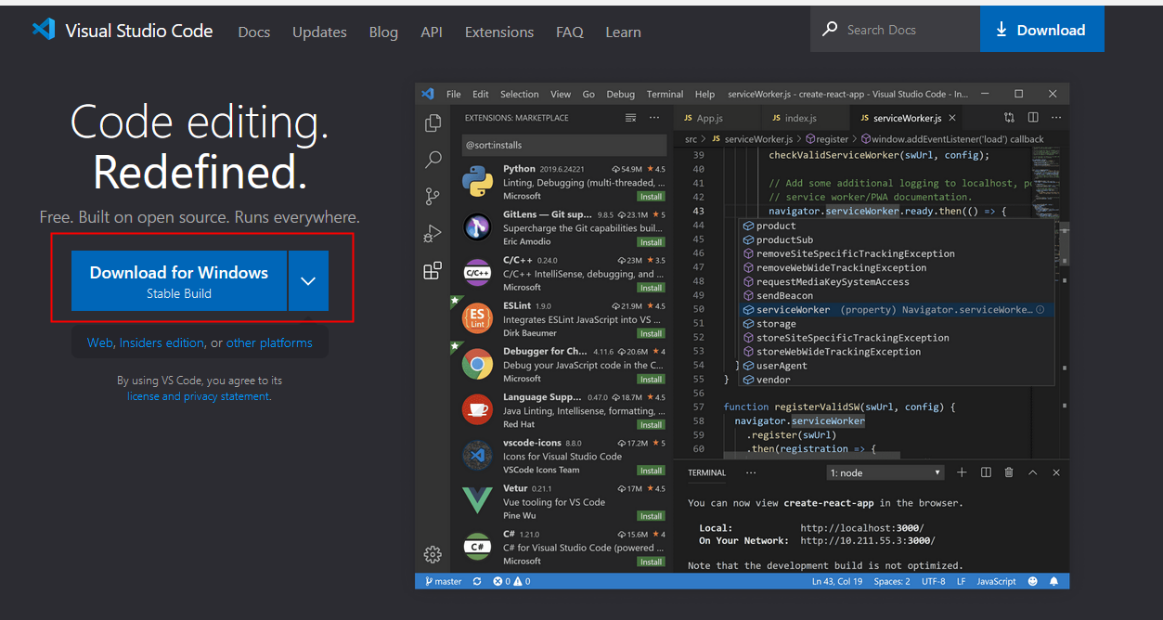


2、输入npm i @vue/cli -g 开始安装Vue-cli脚手架

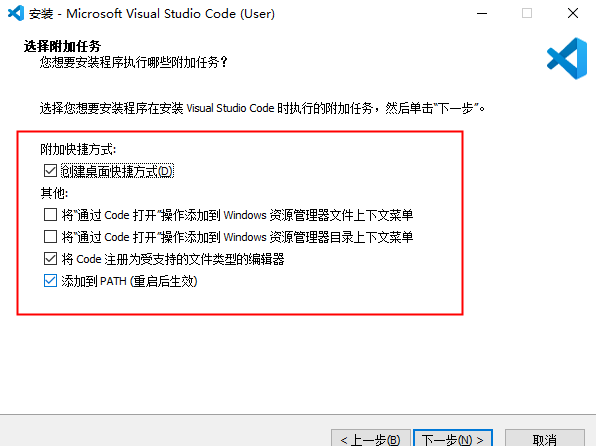


## 安装VSCode开发工具

1、下载Visual Studio Code(以下简称VSCode)安装文件，下载地址为: https://code.visualstudio.com/。



2、下载好之后，双击安装包按提示进行安装即可。安装过程中有几下几个选项需要注意一下：



1、勾选“创建桌面快捷方式”;

2、可选“将Code添加到右键菜单，支持打开文件”；

3、可选“将Code添加到右键菜单，支持打开目录”；

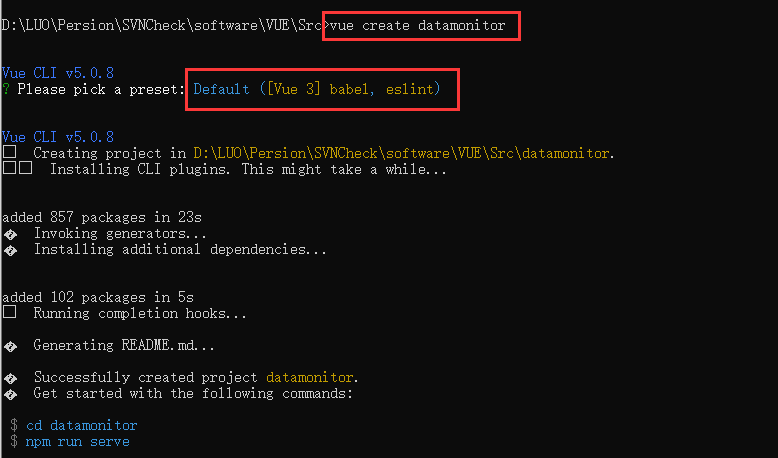
4、不勾选“将Code注册为受支持的文件类型的编辑器”；

5、勾选“添加到PATH”。

# 系统开发与调试

## 创建项目

1、启动一个CMD命令行窗口，运行“vue create datamonitor”命令创建一个VUE项目，选择Vue 3的版本，如下图所示。



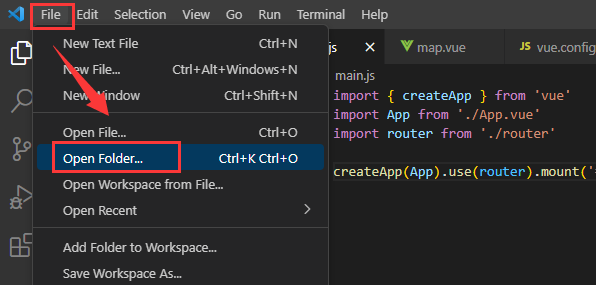
2、创建完成后，datamonitor项目目录结构如下图所示。



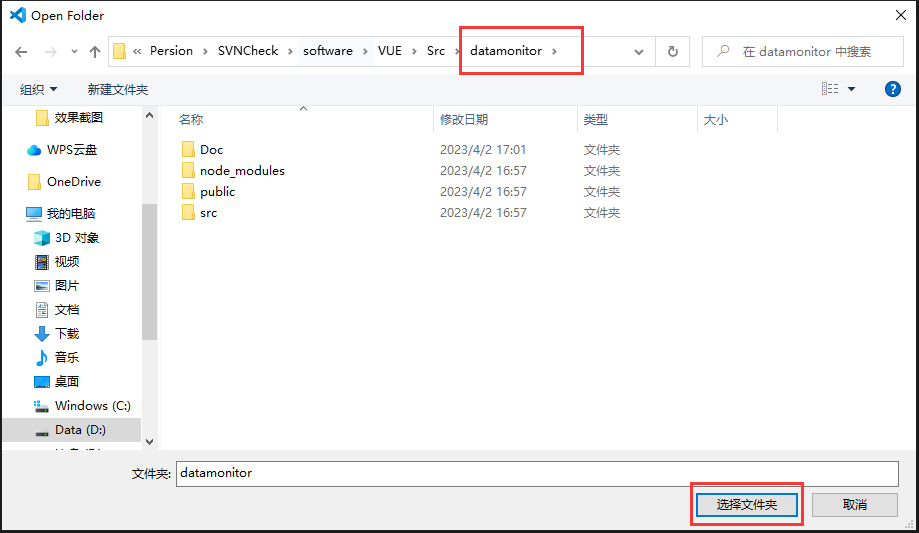
|  |  |
| --- | --- |
| **目录名称** | **用途** |
| build | 项目构建(webpack)相关代码 |
| **dist** | 存放打包之后的代码文件 |
| config | 配置目录，包括端口号等。我们初学可以使用默认的。 |
| node\_modules | npm加载的项目依赖模块 |
| public | 公共静态资源文件 |
| src | 项目源代码 |
| |\_ assets | 静态文件，css、图片等 |
| |\_components | 自定义的vue组件 |
| |\_views | 视图组件 |
| |\_router | 路由配置文件 |
| |\_ store | 与vuex相关 |
| App.vue | 根组件 |
| main.js | 入口文件 |
| package.json | 项目配置文件，包管理、项目名称、版本和命令 |
| vue.config.js | 项目配置信息 |
| **package-lock.json** | 项目包的锁定文件，用于防止包版本不一样导致的错误 |

## 开发代码

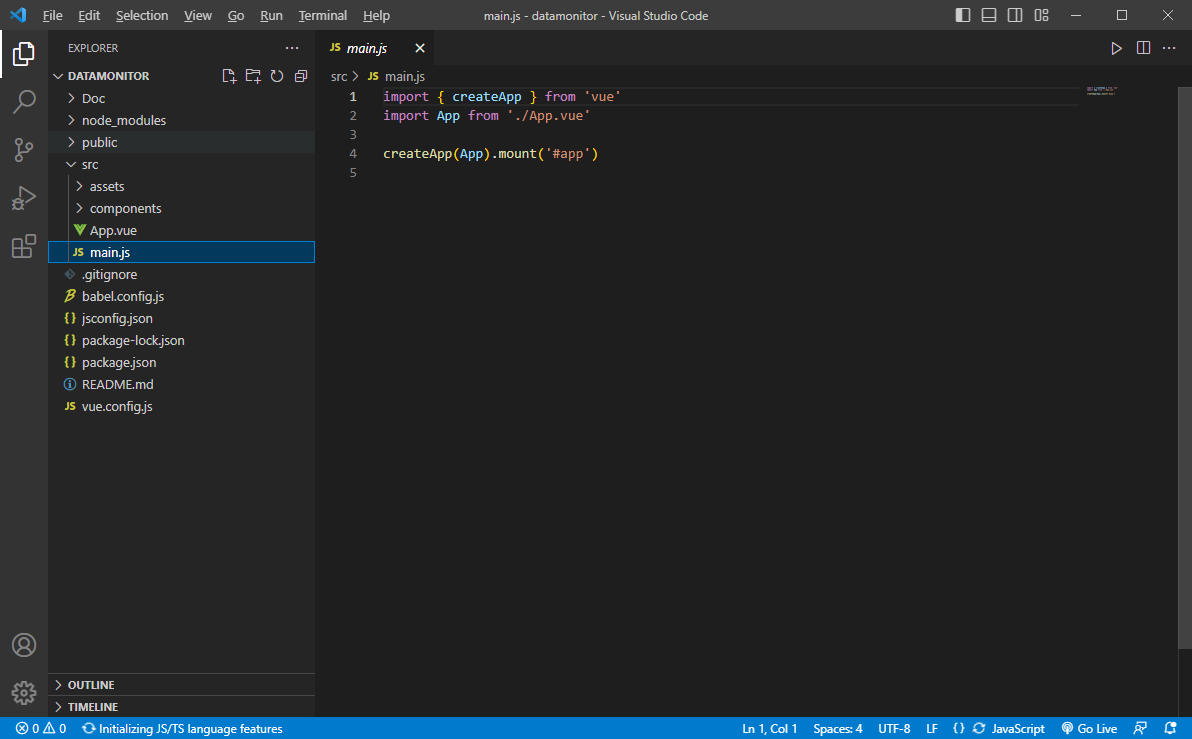
1、启动VSCode程序，点击 [File] -> [Open Folder] 菜单，如下图所示。



2、在弹出的“Open Folder”对话框中，选择datamonitor项目目录，如下图所示。

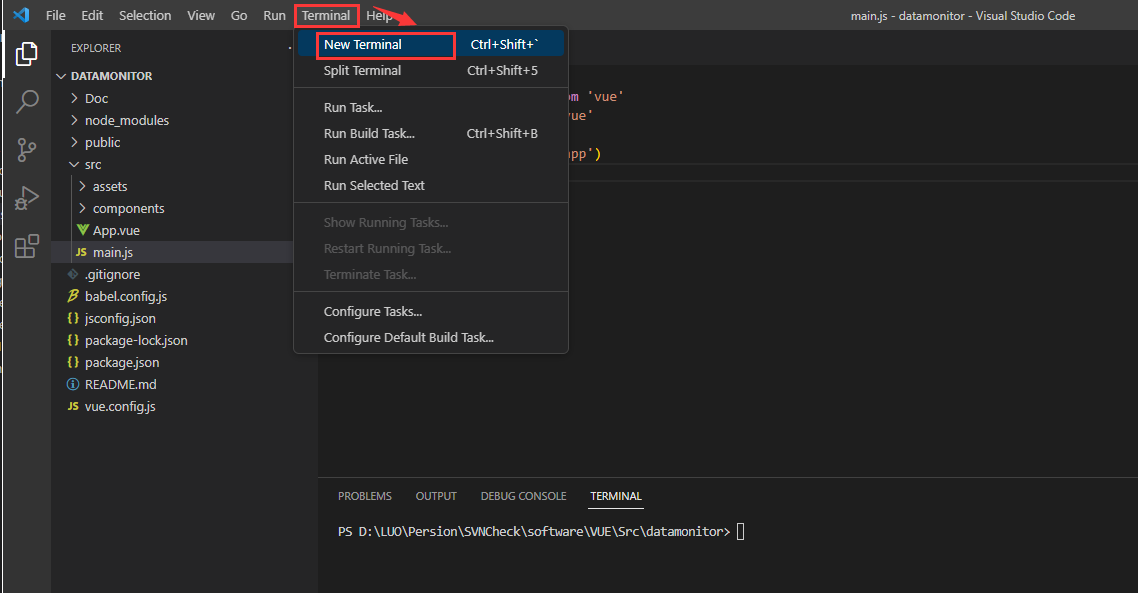


3、成功加载datamonitor项目后，即可在左侧的树形目录中选择需要编辑的代码文件，双击打开进行编写代码即可，如下图所示。

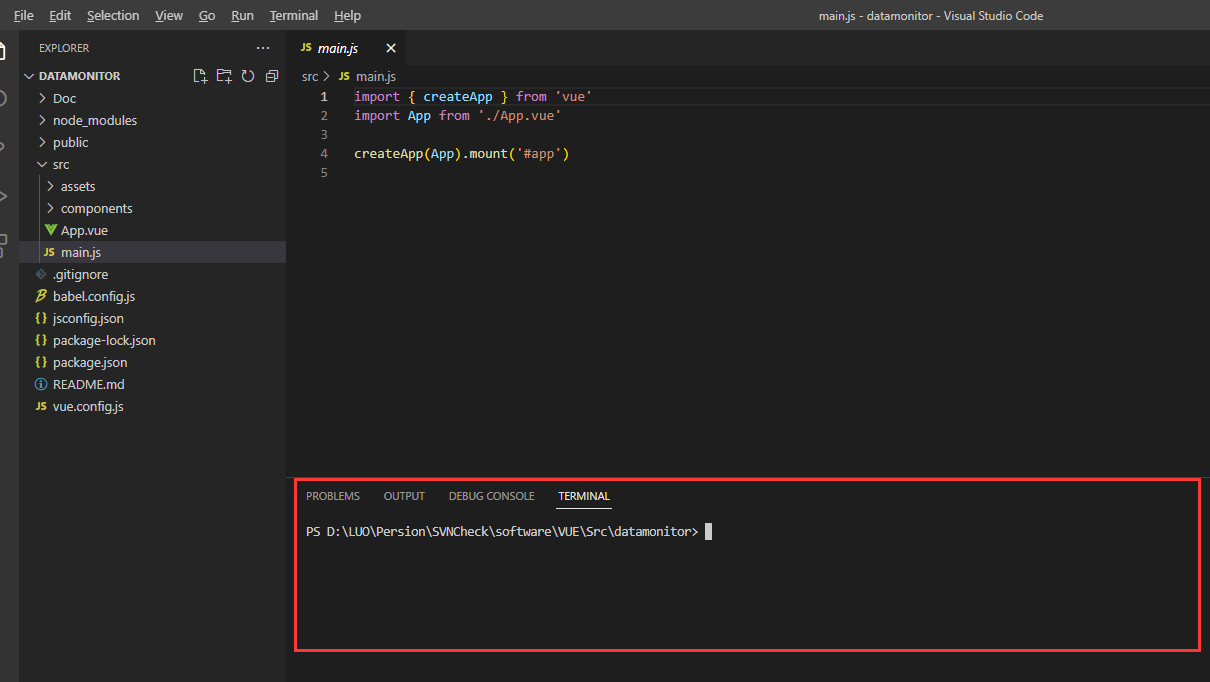


## 运行程序

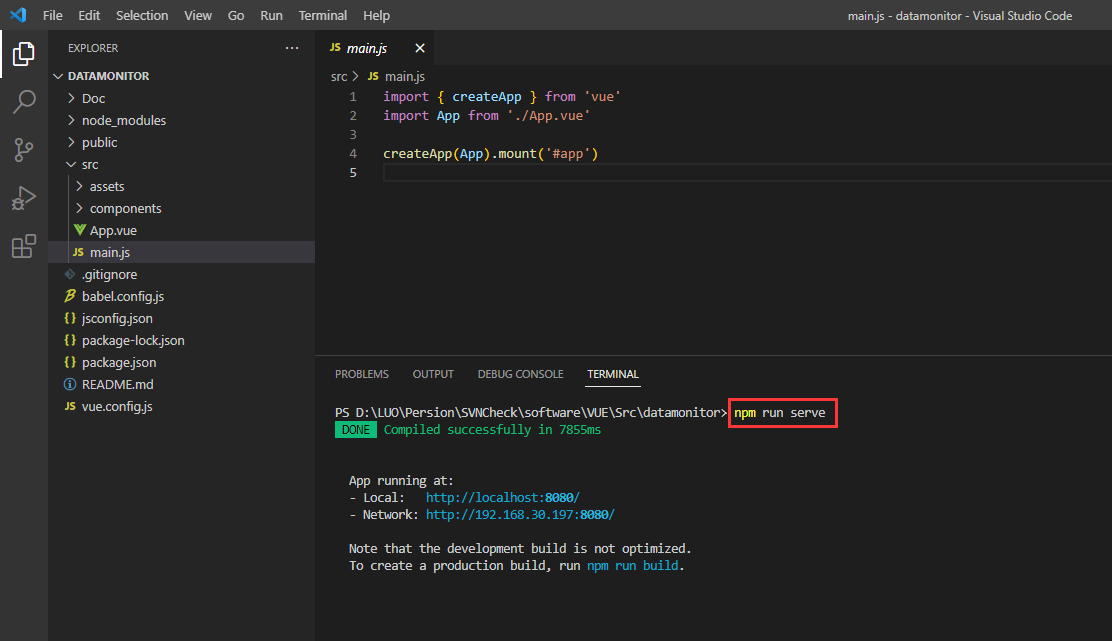
1、点击VScode顶部中的“Terminal”菜单，选择“New Terminal”选项，如下图所示。



弹出命令行窗口，如下图所示。



2、在命令行窗口中，运行“npm run serve”命令，启动程序服务即可，如下图所示。



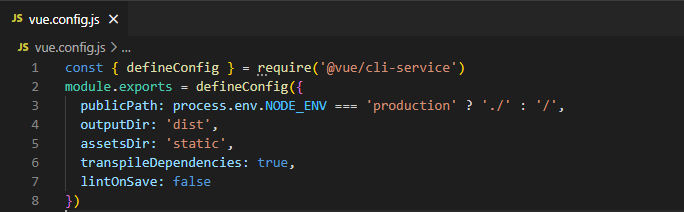
3、正常启动程序服务后，打开浏览器，输入上图中提示的网址：<http://localhost:8080/>，即可访问系统网页，并进行测试调试。

# 系统功能开发

## 配置vue.config.js参数

vue.config.js 是一个可选的配置文件，由于Vue3不再有webpack.config.js配置文件，但是在项目开发中我们可能会需要调整webpack参数, 因此在Vue3的项目当中，我们就需要在根目录创建vue.config.js去完成webpack的一些特殊配置，默认它会被 @vue/cli-service 自动加载。

请按照下图中的内容配置vue.config.js的基本参数，如下所示。



const { defineConfig } = require('@vue/cli-service')

module.exports = defineConfig({

  publicPath: process.env.NODE\_ENV === 'production' ? './' : '/',

  outputDir: 'dist',

  assetsDir: 'static',

  transpileDependencies: true,

  lintOnSave: false

})

publicPath：部署应用包时的基本 URL；

outputDir：build 时输出的文件目录；

assetsDir: 放置静态文件夹目录

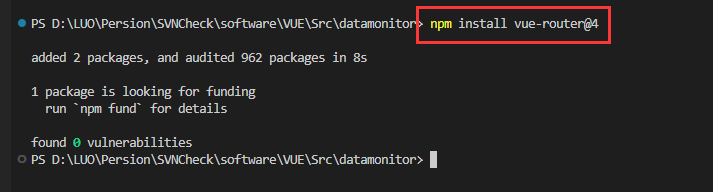
lintOnSave：是否在保存的时候使用eslint-loader进行检查。 有效的值：ture | false|error，当设置为error时，检查出的错误会触发编译失败。

transpileDependencies：默认情况下 babel-loader 会忽略所有 node\_modules 中的文件。如果你想要通过 Babel 显式转译一个依赖，可以在这个选项中列出来。

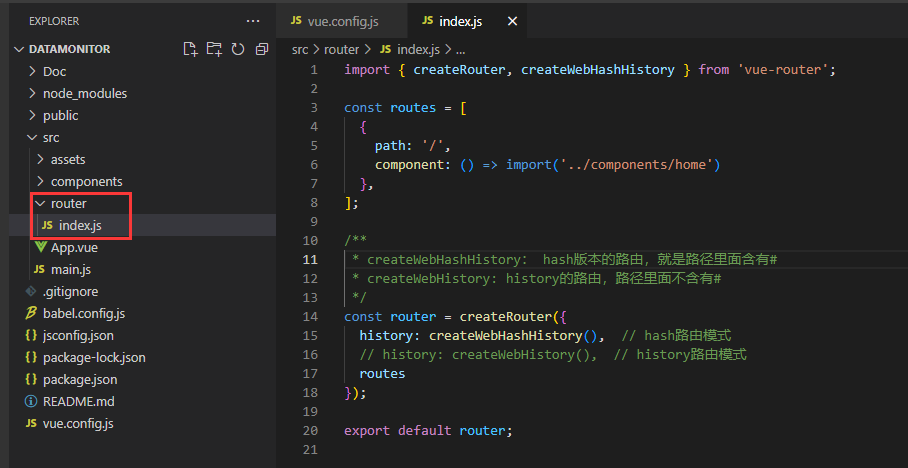
## 添加Vue Router路由

Vue Router是Vue.js的官方路由，它与Vue.js核心深度集成，让用Vue.js构建单页应用变得非常容易。vue-router是基于路由和组件的，路由用于设定访问路径, 将路径和组件映射起来，目前Vue路由最新的版本是4.x版本。

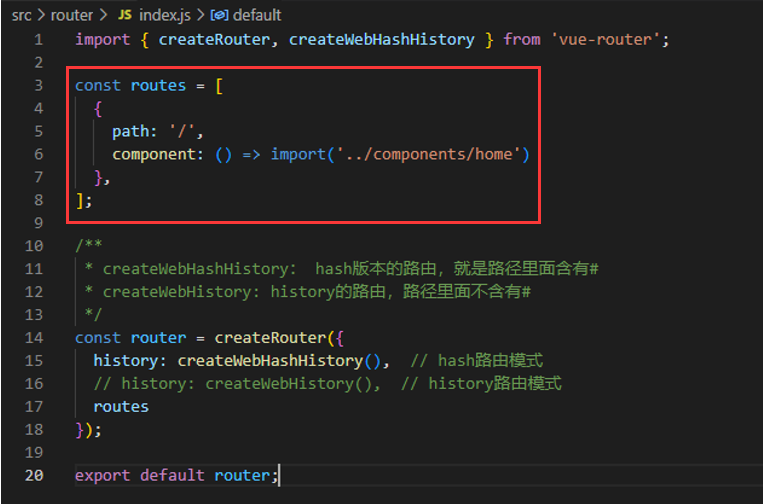
1、安装Vue Router：启动一个CMD命令行窗口，运行“npm install vue-router@4”开始安装Router。



2、在项目src目录下创建router子目录，然后在router子目录下创建index.js代码文件，如下图所示。



3、在index.js代码文件中，添加如下图所示的代码内容。



上图中方框中的代码为路由配置的映射关系，上图中的示例代表将当前的访问根路径“/” 映射到components目录下的home.vue组件文件。

import { createRouter, createWebHashHistory } from 'vue-router';

const routes = [

  {

    path: '/',

    component: () => import('../components/home')

  },

];

/\*\*

 \* createWebHashHistory： hash版本的路由，就是路径里面含有#

 \* createWebHistory: history的路由，路径里面不含有#

 \*/

const router = createRouter({

  history: createWebHashHistory(),  // hash路由模式

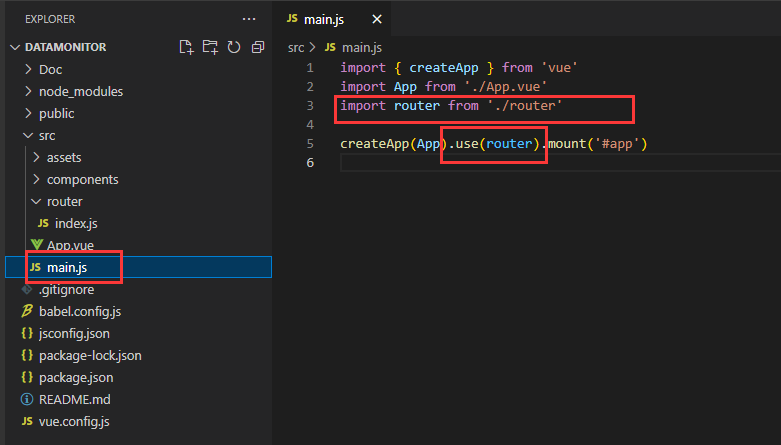
  // history: createWebHistory(),  // history路由模式

  routes

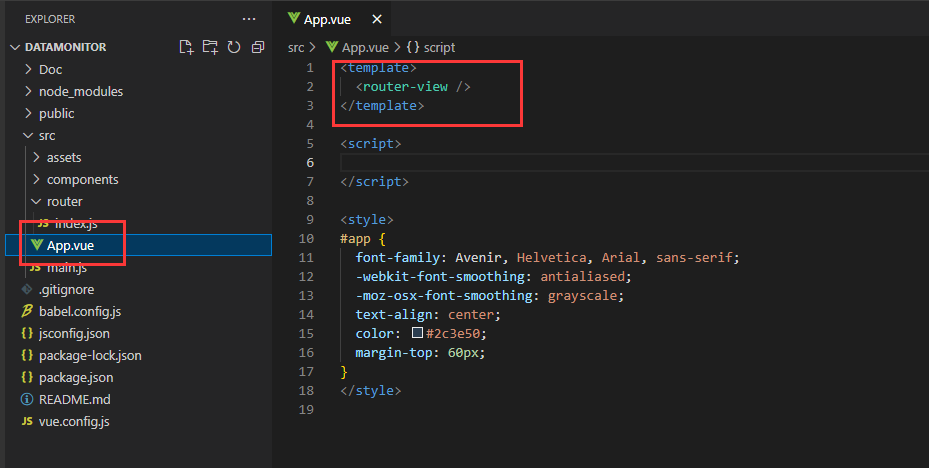
});

export default router;

4、router的index.js文件创建好之后，还需要在main.js文件中通过createApp.use()方法引入，代码如下图所示。



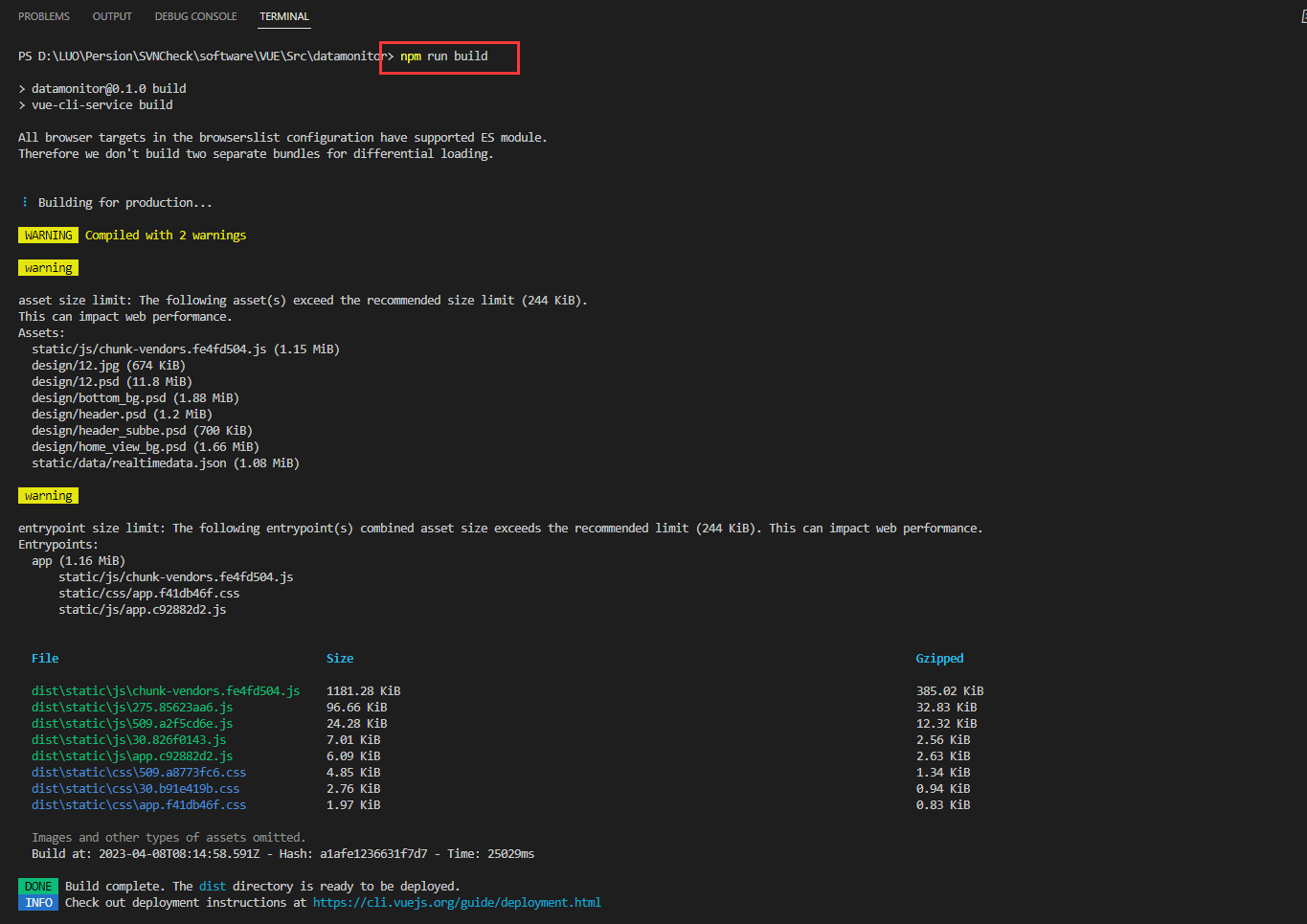
5、最后，在项目根组件App.vue文件中添加router使用代码即可，如下图所示。



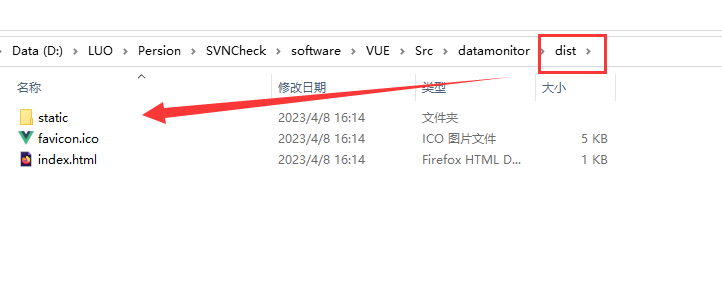
# 系统发布

系统开发完成后，可按以下操作步骤进行发布。

1、在VSCode中的命令行窗口中，运行“npm run build”命令，启动程序发布服务即可，如下图所示。



2、完成发布命令执行后，系统将打包发布好的程序存放在“dist”目录，如下图所示。



3、将dist目录中的所有发布程序文件复制到Web服务器中进行部署即可。