# 前端的运行

# 前端开发环境

## NodeJs简介

NodeJS 是基于Chrome V8引擎的 JavaScript 运行环境。NodeJS使用事件驱动，非阻塞型I/O。NodeJS的包管理生态是 NPM，是现在世界上最大的开源程序包库。JavaScript是一门脚本语言，它需要一个运行环境。就好像PHP需要Apache、JSP需要Tomcat、Java需要JVM等等， 而NodeJS之前，JavaScript运行环境是浏览器，也就是JavaScript在网页中才能跑起来。NodeJS之后JavaScript又多了一个运行环境，就是NodeJS。

## NodeJs安装

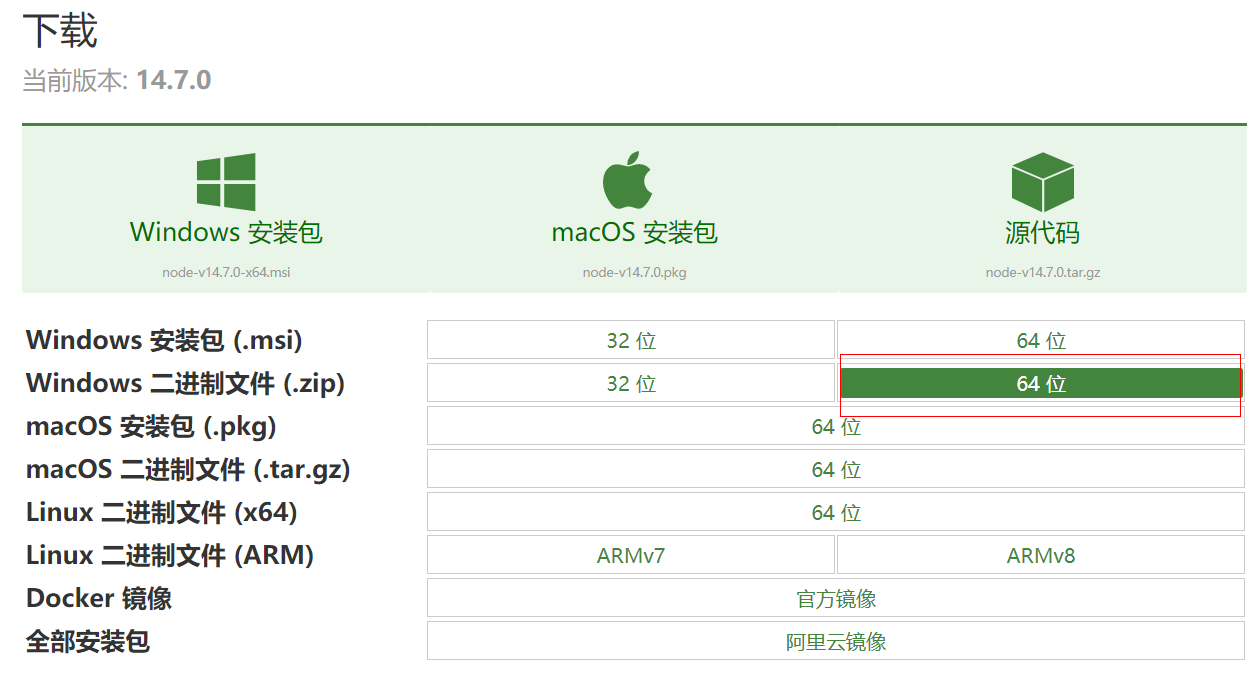
要使用NodeJs首先要安装它。

### 下载Node.js

浏览器访问：

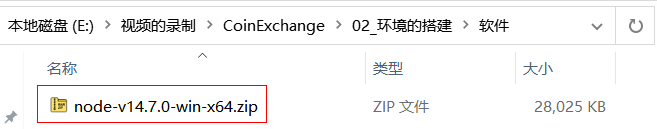
|  |
| --- |
| http://nodejs.cn/download/ |

我们可以看见现在nodejs现在最新的版本为 **14.7.0，**根据你的需要选择要下载的nodejs的版本。**Windows用户推荐使用二进制文件**，也就是**zip**的安装方式。现在我们下载64位的nodejs:



点击下载完毕即可。

当然，我们也为你准备了本地的安装包文件，你可以从本章的软件目录里面去下载它。



### 解压Node.js

将下载的nodejs复制到你的安装软件的目录里面。以我的开发目录为例子，首先我切换到devtools里面：



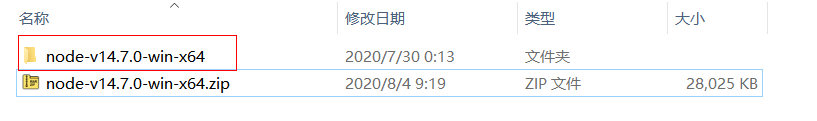
在该文件夹里面新建一个nodejs文件夹：



将下载的nodejs的压缩包（zip文件）复制到nodejs文件夹里面：



现在，使用压缩软件解压就可以了。



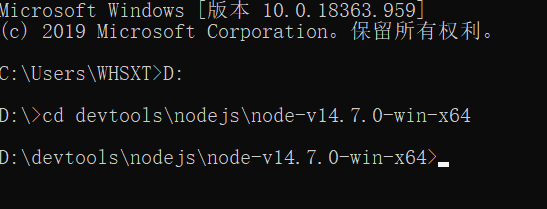
### 配置环境变量

在上面一步我们已经将nodejs的压缩包解压了，现在，我们来尝试运行它。

打开 终端（win+R），切换到该目录里面：

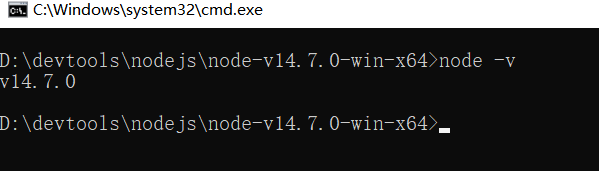
输入以下命令：

|  |
| --- |
| D:  cd devtools\nodejs\node-v14.7.0-win-x64\ |

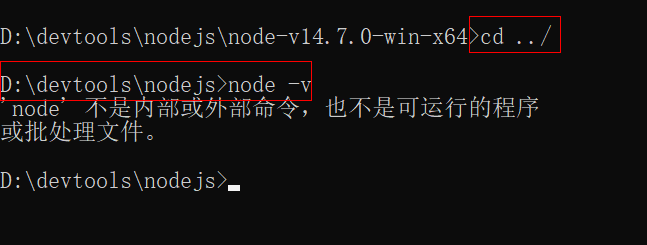


尝试运行：

|  |
| --- |
| node -v |



发现已经成功的运行了。但是，现在我们只能在该目录下运行node，若切换到别的目录，就无法运行了。比如：



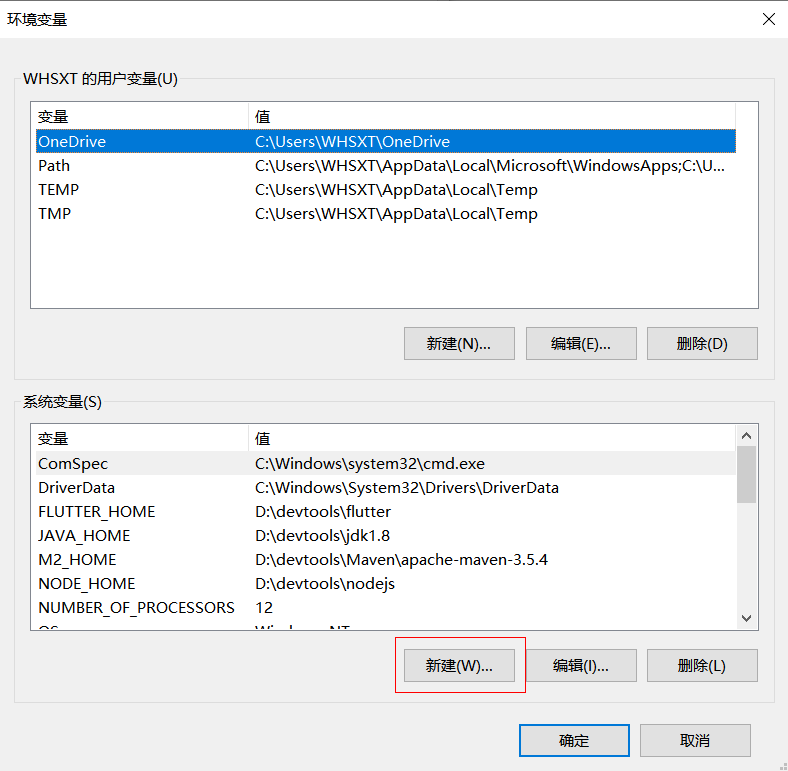
当我们切换到外面一级，发现已经运行不了，聪明的你肯定已经想到，需要**配置环境变量**了。

打开控制面板--->系统和安全--->系统

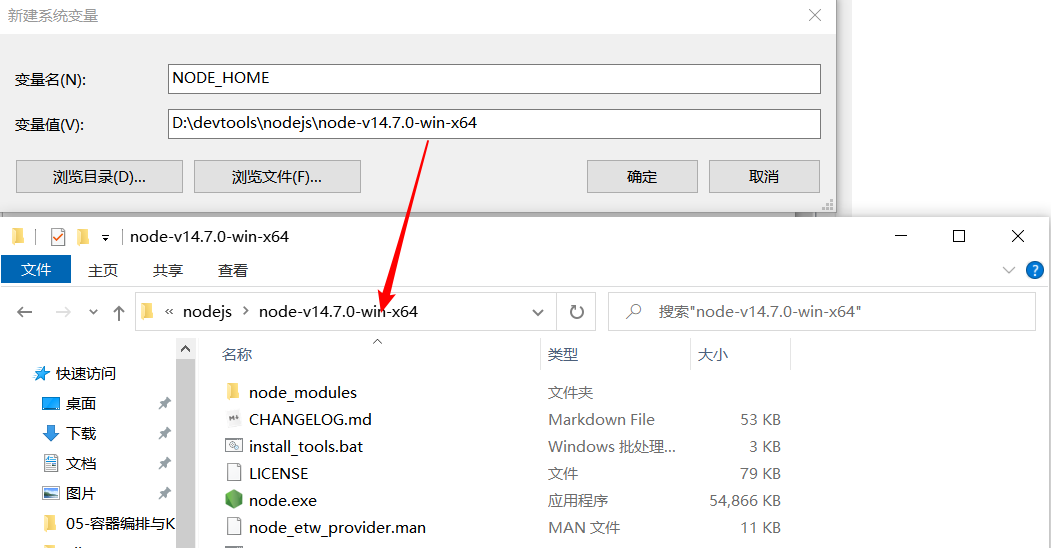




点击环境变量：

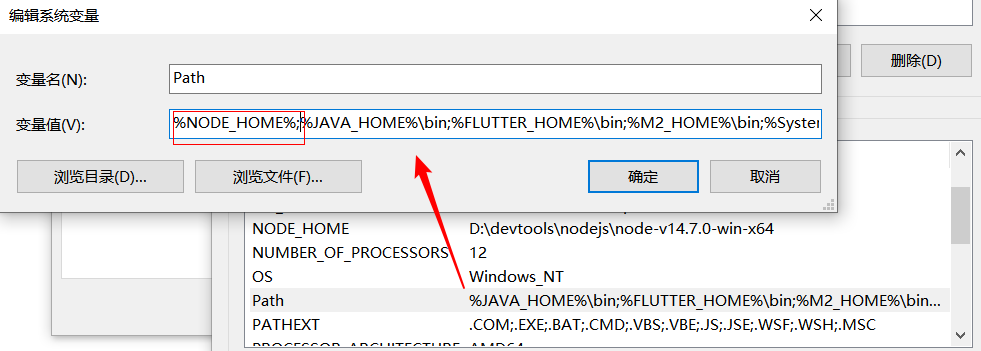


点击新建，我们先来创建一个**NODE\_HOME**的变量值：



* 变量名：NODE\_HOME
* 变量值：D:\devtools\nodejs\node-v14.7.0-win-x64 该值就是你解压nodejs后的目录

现在，我们修改PATH，让windows能从该目录里面寻找指令：

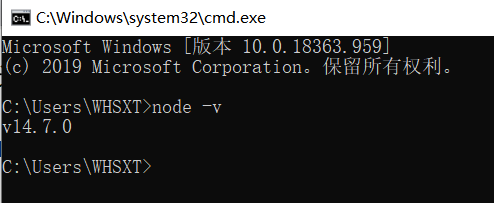


我们点击Path后，将光标移动到变量值的最前面，添加 **%NODE\_HOME%;就可以了。**

**注意，仅仅是%NODE\_HOME%，~~不是%NODE\_HOME%bin。~~**

完成后点击确定即可。

现在，你可以不用切换目录来运行我们的node了。



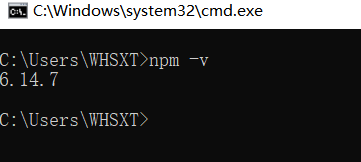
## 镜像加速

### npm 介绍

NPM是随同NodeJS一起安装的包管理工具，能解决NodeJS代码部署上的很多问题，常见的使用场景有以下几种：

* 允许用户从NPM服务器下载别人编写的第三方包到本地使用。
* 允许用户从NPM服务器下载并安装别人编写的命令行程序到本地使用。
* 允许用户将自己编写的包或命令行程序上传到NPM服务器供别人使用。

由于新版的nodejs已经集成了npm，所以之前npm也一并安装好了。同样可以通过输入 "npm -v" 来测试是否成功安装。命令如下，出现版本提示表示安装成功:



### cnpm的安装和说明

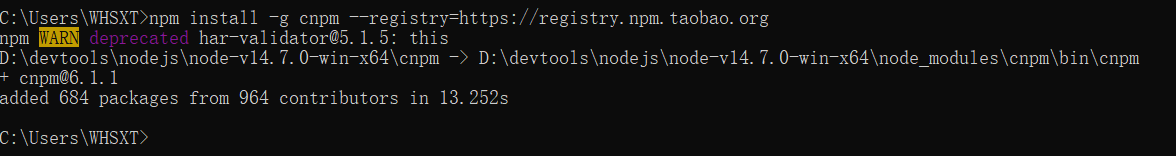
Npm 包服务器在国外，导致国内下载第三方的包异常的慢，在此，我们可以使用淘宝提供给我们的加锁地址。看说明：



直接运行：

|  |
| --- |
| npm install -g cnpm --registry=https://registry.npm.taobao.org |

等待片刻，当出现：



代表你已经安装成功了，以后，你可以使用cnpm 代替npm 来下载第三方的应用了。

# 前端开发软件

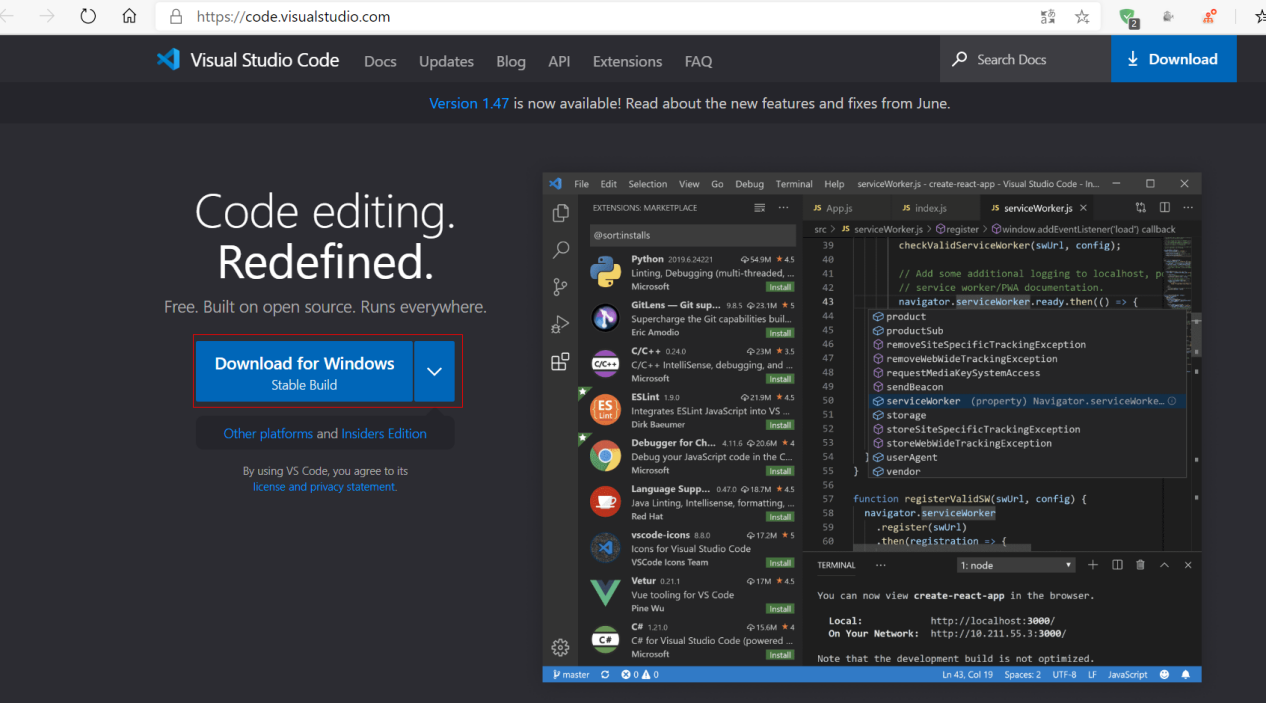
前端开发软件我们使用的是vscode。Vscode是一款轻量且强大的跨平台开源代码编辑器（IDE），支持Windows，MacOS 、Linux。内置JavaScript、TypeScript和Node.js支持，而且拥有丰富的插件生态系统，可通过安装插件来支持Java、Python、C++、PHP等其他语言。

## 2.1 vscode下载

浏览器打开：

|  |
| --- |
| https://code.visualstudio.com/ |

点击Download for Windows 即可下载：

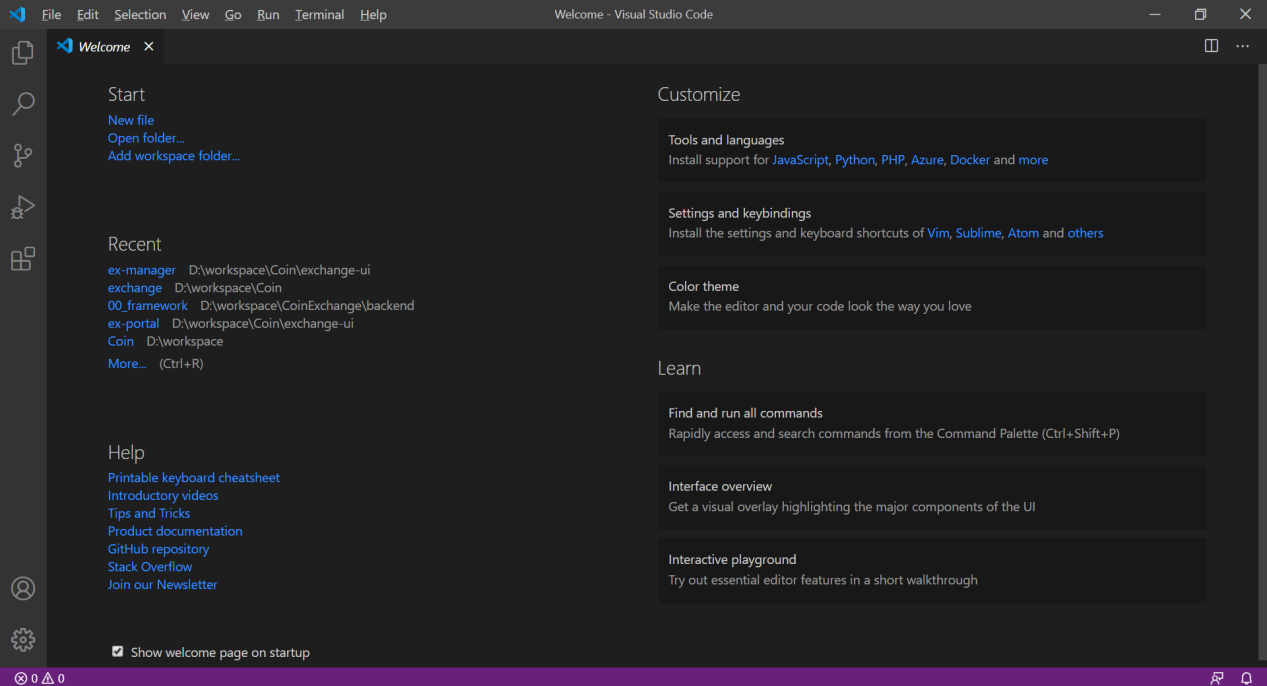


当然，你也可以从本章的软件里面去下载它。



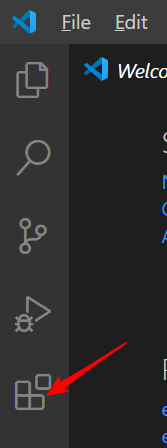
## 2.2 vscode的安装

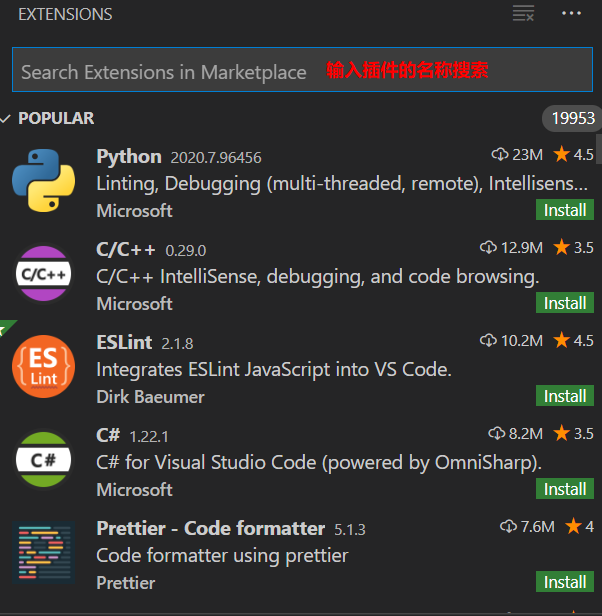
双击下载的文件，一直点击下一步就可以完成安装了。安装完成后，运行软件，如下图所示：



## 2.3 插件的安装

点击该按钮：

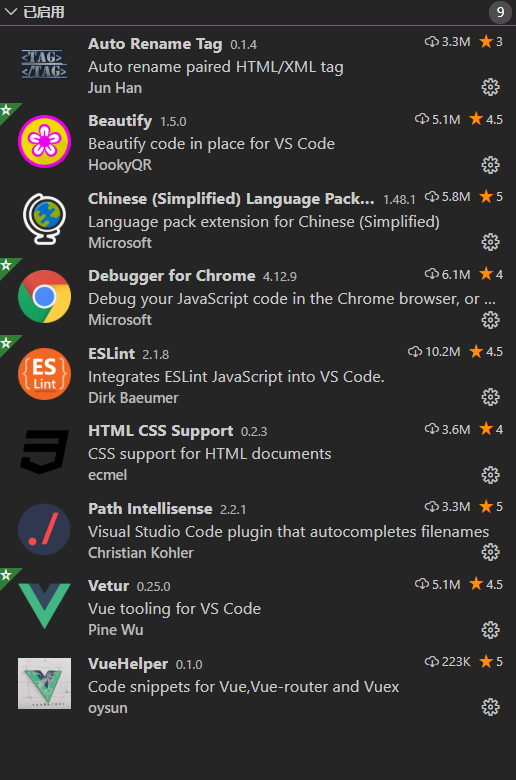




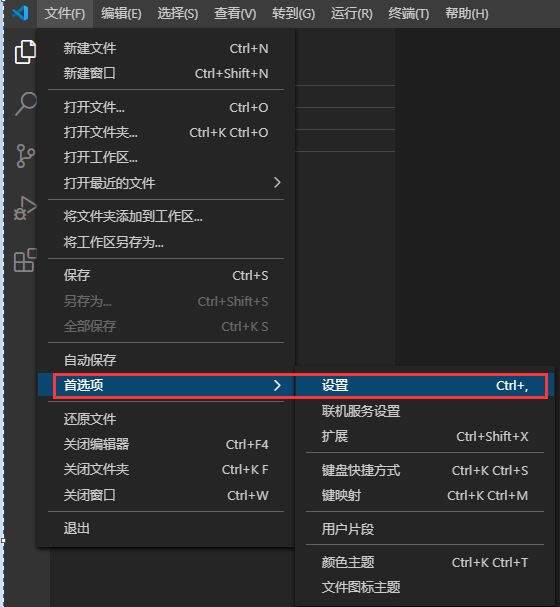
以下是开发vue 是推荐安装的插件：

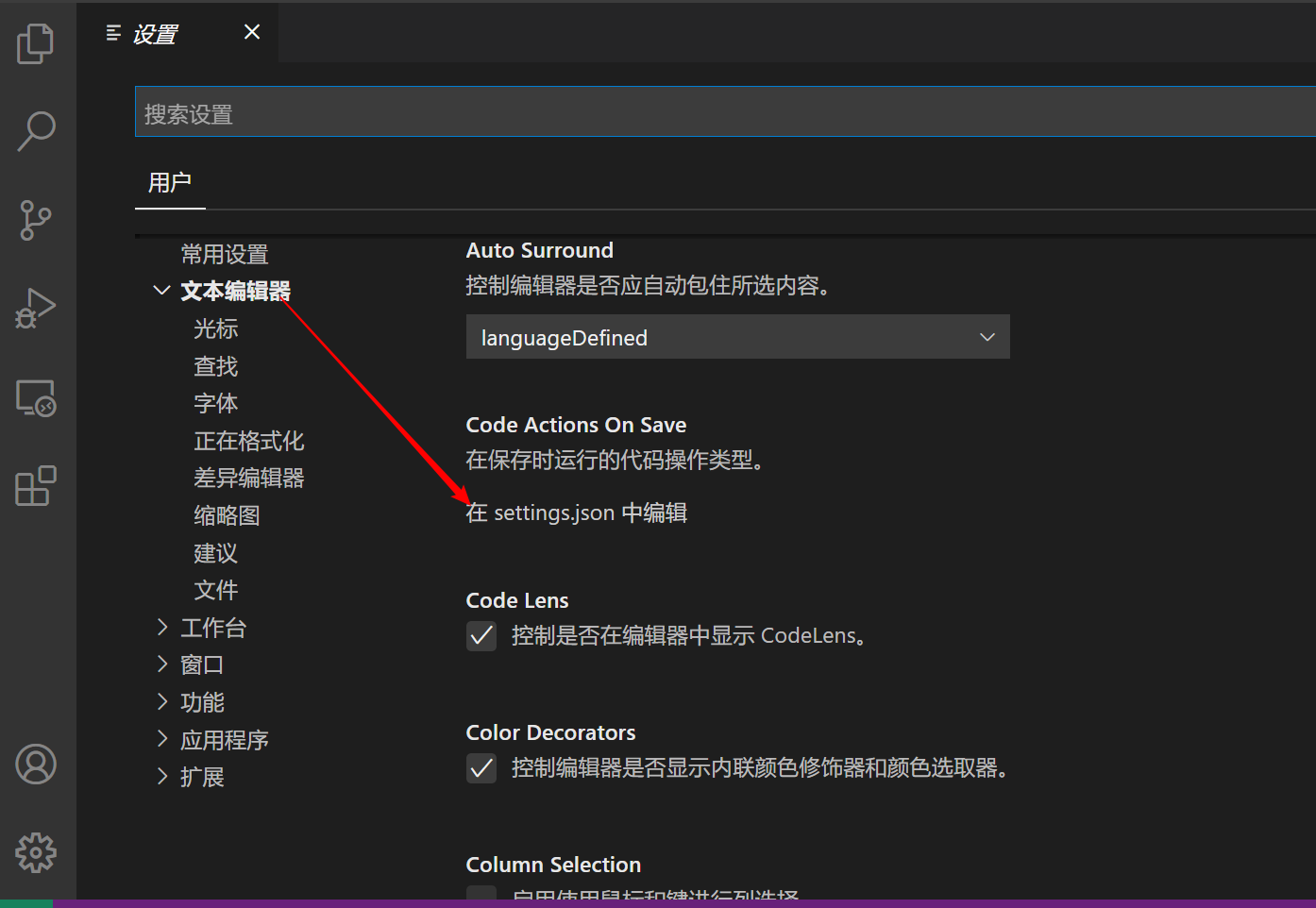
|  |  |  |
| --- | --- | --- |
| 插件名称 | 说明 | 是否必须 |
| Chinese | 中文语言支持 | 是 |
| vetur | Vue 开发工具 | 是 |
| Vuehelper | Vue、Vue-router、Vuex代码提示 | 是 |
| eslint | Javascript代码检测工具 | 是 |
| Debugger for Chrome | 代码调试工具 | 否 |
| HTML CSS Support | Css代码提示 | 否 |
| Path Intellisense | 自动补全路径的 | 否 |
| Beautify | 代码格式化 | 否 |
| Auto Rename Tag | 自动完成尾部闭合标签修改 | 否 |

安装完毕后，记得重启，重启完成后，如下图：



## 2.4 配置的导入





点击在settings.json中编辑。

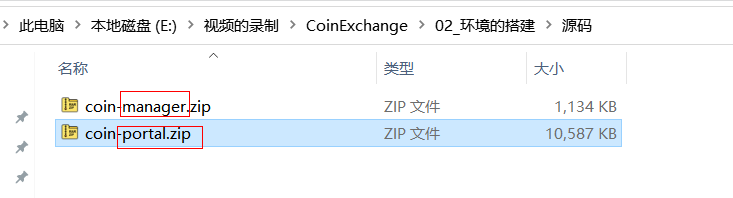
切换到今天的软件目录，打开【vscode配置文件.json.txt】，将内容复制进去即可。

# 前端的运行

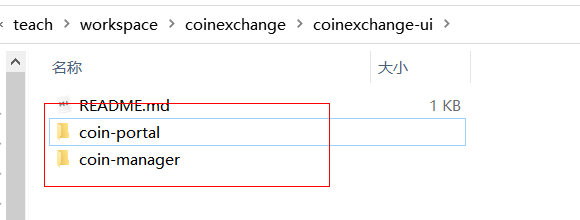
前端包含：**后台管理系统、前台系统、移动端全栈系统**。

前端所有的功能都已经开发完毕了。在本项目里面，你可以**选择自己构建交互逻辑**，或者直接使用我们**提供的代码**。在以后的授课时，我们不会写前端代码，但我们会分析所有的核心的前端代码，以帮助你快速的上手前端系统和理清和后端的交互逻辑。

## 3.1 下载前端的文件



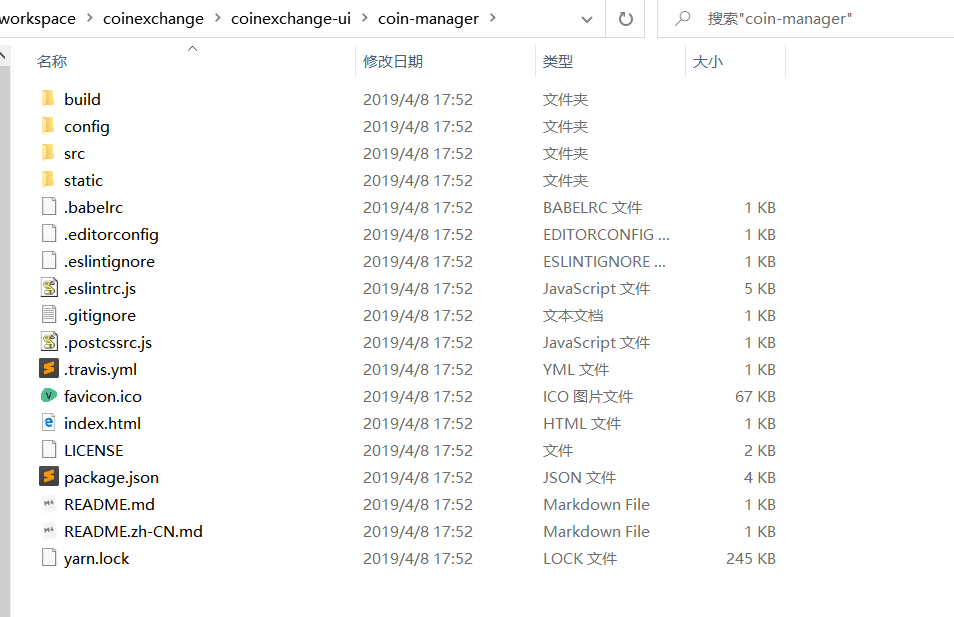
## 3.2 解压到我们的git仓库里面



**注意，在仓库里面不需要将压缩文件放在里面。**

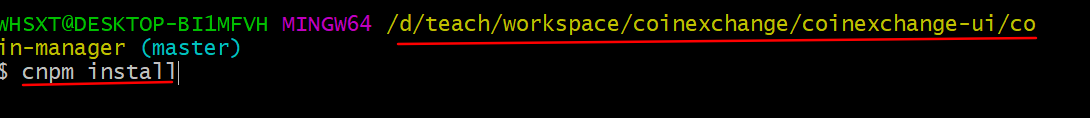
## 3.3 后台管理系统的运行

进入到coin-manager里面：



安装依赖，执行如下的命令：

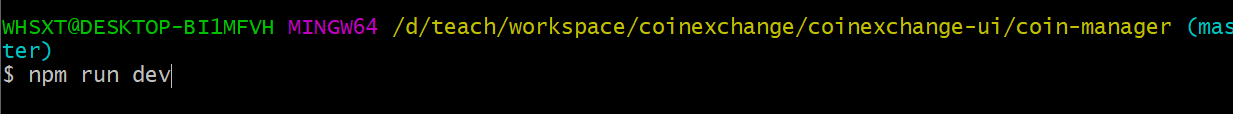
|  |
| --- |
| cnpm install |



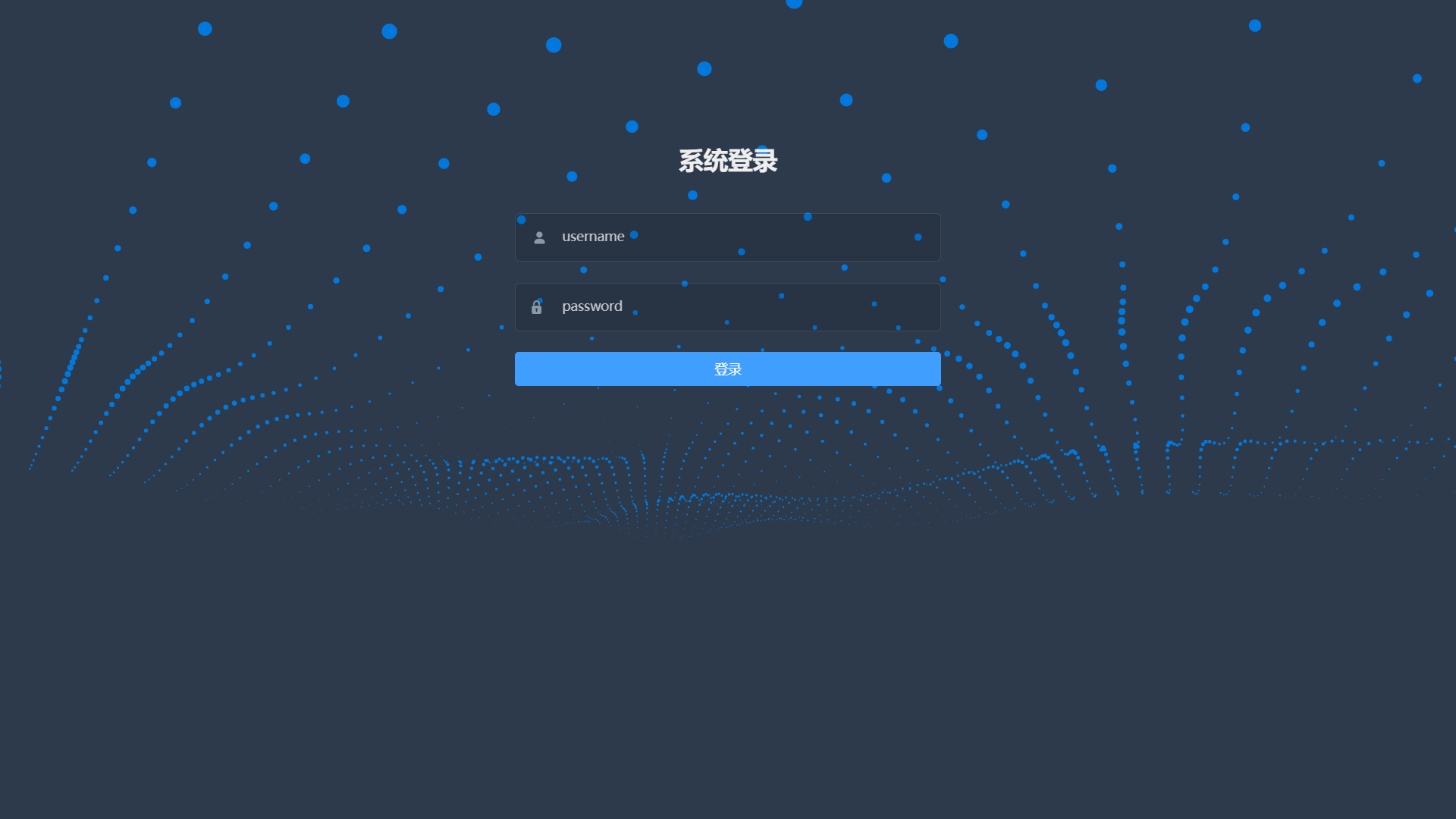
执行介绍代码依赖安装完毕。

运行项目：

|  |
| --- |
| npm run dev |



运行成功后：



## 3.4 前台系统的运行

进入到：

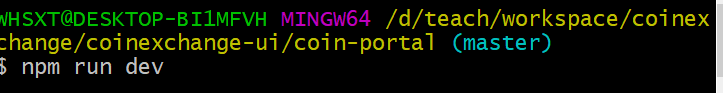


该目录里面，打开gitbash，执行：

|  |
| --- |
| cnpm install |

执行成功后，运行项目：

|  |
| --- |
| npm run dev |



运行成功后：

