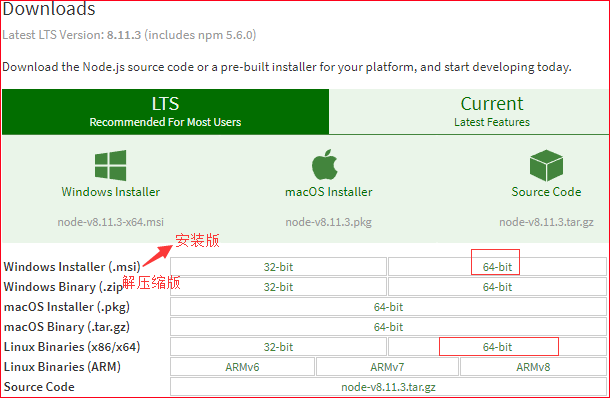
Nodejs

# 一、Nodejs安装

## 1. window上安装

Nodejs下载地址：https://nodejs.org/en/; 可以选msi版本,也可以选zip解压缩版本,在此选择zip版;如图所示:



下载完成,解压缩zip压缩包,放到自己指定的文件夹,添加到环境变量;

## 2. Linux 上安装

# 安装步骤

1. wget <https://nodejs.org/dist/v8.11.3/node-v8.11.3-linux-x64.tar.xz>

2. xz –d node-v8.11.3-linux-x64.tar.xz

3. tar –xvf node-v8.11.3-linux-x64.tr

4. mv node-v8.11.3-linux-x64 /usr/local/node-v8

5. vim /etc/profile 加入环境变量PATH=$PATH:/usr/local/node-v8/bin

6. 保存并 source profile 使之立即生效

7. node –v 测试是否安装成功

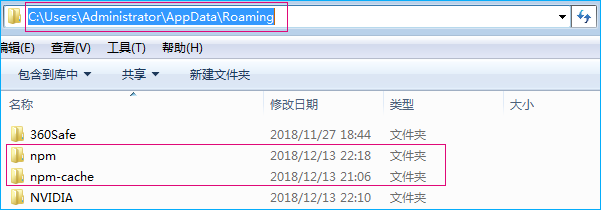
# 二、npm 安装及使用

## 1. 默认安装的npm

安装完成node之后,查看node安装目录会发现npm.cmd程序。其实安装node的过程已经帮我们安装了npm包版本管理工具。使用命令: npm –v 查看当前npm版本

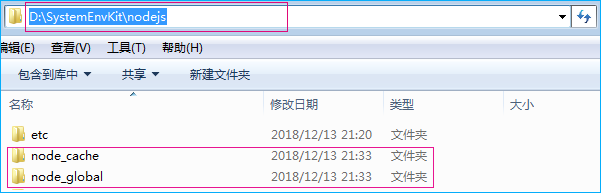
C:\Users\Administrator\Desktop\npm-v.png

npm默认仓库路劲为: C:\Users\Administrator\AppData\Roaming 下,如图所示:



## 2. 改变cache和仓库目录

在node安装目录下新建两个目录,如图所示:

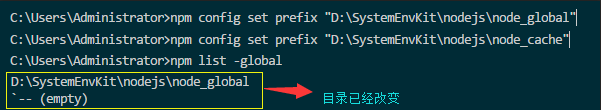


运行以下两条命令配置npm地址:

$ npm config set prefix “D:\SystemEnvKit\nodejs\node\_global”

$ npm config set cache “D:\SystemEnvKit\nodejs\node\_cache”

如图所示:



npm list –global显示当前全局目录所在位置,以后的全局安装-g都会指向该命令显示的目录。

## 3. 配置远程镜像仓库

官方地址:

npm官方地址：<https://www.npmjs.com/>

npm官方镜像：<https://registry.npmjs.org/>

由于npm的地址都在国外,所以下载速度比较慢,这里我们需要配置国内代理的镜像仓库,用于加速下载。

国内镜像:

淘宝npm镜像

* 搜索地址：<http://npm.taobao.org/>
* registry地址：<http://registry.npm.taobao.org/>

cnpmjs镜像

* 搜索地址：<http://cnpmjs.org/>
* registry地址：<http://r.cnpmjs.org/>

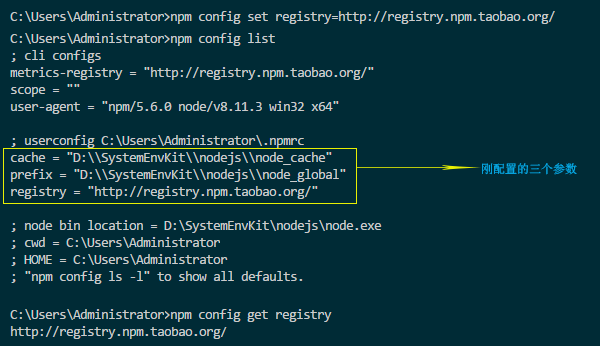
相关命令：

$ npm config set registry=http://registry.npm.taobao.org/

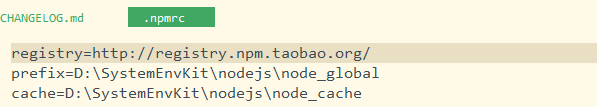
$ npm config list # 该命令可以显示所有配置信息

$ npm config get registry # 该命令查询当前npm的仓库地址

如图所示：



注意有一个 .npmrc文件,该文件是npm的用户配置文件,按如图所示的路径我们可以找到它并查看它。如图所示:

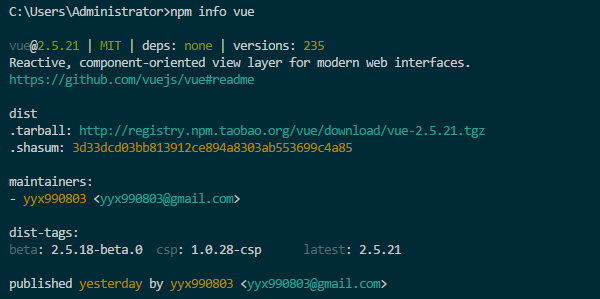


可以看到与我们手动配置的信息一致。

测试一下配置的镜像地址是否可行,命令如下：

$ npm info plugins-name # plugins-name代表要查看的模块名，比如vue

如图所示：



## 4. 升级NPM

由于我们的npm版本并不是最新的我们需要升级npm版本,升级版本命令:

$ npm install npm –g # -g 代表全局按照,即安装到node\_global目录下

如图所示：



查看如果还是没有升级到最新版本,我们可以采用如下安装命令：

$ npm install npm@latest –g # @latest 指定更新最新版本

## 5. npm的环境变量配置

当以全局方式升级npm之后,默认的模块D:\SystemEnvKit\nodejs\node\_modules 目录将会改变为 D:\SystemEnvKit\nodejs\node\_global\node\_modules ,即升级后npm被安装到了全局目录node\_global下,所以再直接运行npm install等命令会报错的。

所以我们需要增加环境变量：

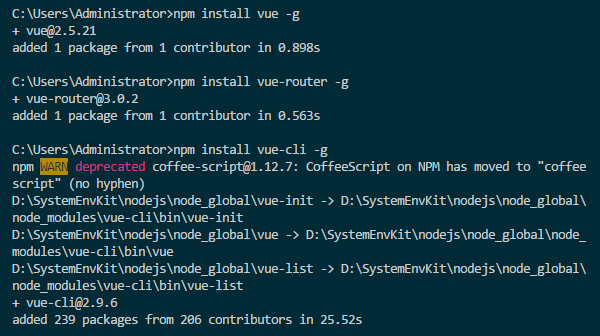
* 新增系统环境变量

NODE\_PATH = D:\SystemEnvKit\nodejs\node\_global\node\_modules

* PATH环境变量追加路径

D:\SystemEnvKit\nodejs\node\_global

测试安装vue如图所示：



显示都安装成功,可以进入node\_global下查看对应的文件。

## 6. npm包本地安装与全局安装

有两种方式用来安装npm包:本地安装和全局安装。至于选择哪种方式来安装,取决于我们如何使用这个包。

本地安装:本地安装的包只会被下载到指定目录下的node\_modules目录下；对应不同的项目,功能不同,我们需要的依赖包也不尽相同,所以只需要本地安装即可, 这种方式也是 npm install 命令的默认行为。

全局安装:如果我们需要将包作为一个命令行工具那就应该选择全局安装;全局安装的包会被下载到nodejs下配置的node\_gobal目录下,可以让我们在任何目录下使用这个包。比如express,vue相关我们都可以选择全局安装。

本地安装与全局安装命令:

# @scope :指定域

# @name :安装模块名称

# @tag :标签 比如：@latest

# @version :指定版本

$ npm install [<@scope>/]<name>[<@tag>|<@vresion>]

$ npm install [<@scope>/]<name>[<@tag>|<@vresion>] –g # -g代表全局安装

## 7. package.json文件管理本地安装

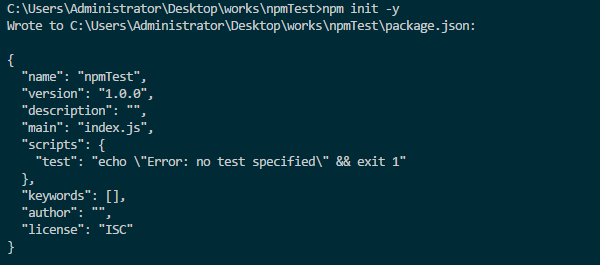
package.json文件就好比maven中pom.xml文件一样,通过该文件,可以轻松管理我们需要安装的依赖包。

创建 package.json文件:

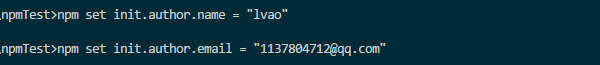
* 以问答形式:即命令行运行npm init 根据提示一步一步的填写信息生成;
* 生成默认文件: 运行 npm init [--yes|-y] 生成一个默认的package.json。

如图所示:



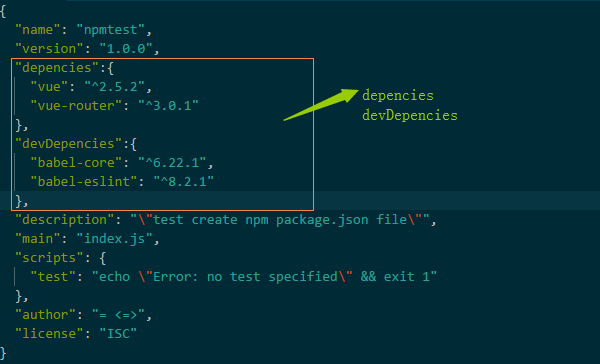


默认的author为空,可以通过以下命令设置：



初始生成的package.json中是没有依赖属性的(depencies) 。

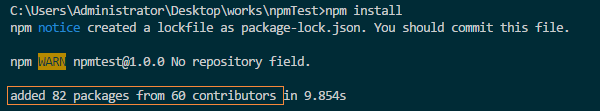
手动编辑package.json文件:

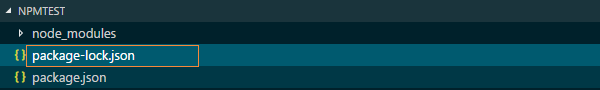


添加depencies与devDepencies 之后,命令行执行npm install就会安装对应的依赖包到当前项目下的node\_modules目录下,如果该目录不存在,则会创建这个目录,并且还会生成一个package-lock.json文件;如果这些依赖包已经存在,在没有package-lock.json文件的情况下,npm install就会更新这些已经存在的包到最新版本。

package-lock.json 是对所有安装依赖的详细描述,是记录当前状态下实际安装的各个npm package的具体来源和版本号。使用此文件,能让我们锁定依赖包的版本,即我们通过当纯的npm install 命令是没有办法更新已安装的依赖包。如果我们想更新某个包的版本,我们需要使用 npm install name@x.x.x 这样去更新指定的包和版本。

如图所示:

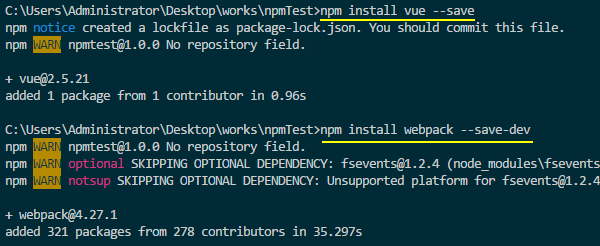




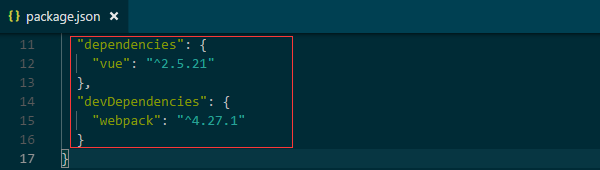
通过--save和--save-dev将安装的包依赖信息写入到package.json中的dependencies和devDependencies属性。

在开发的过程中,我们经常会使用我们事先没有在package.json中指定的依赖包,此时我们安装该依赖包时,就可以在npm install packName 中使用 --save 或 --save-dev 参数同时安装该依赖包并将包信息加入到package.json的的依赖属性中去。

如图所示:



安装之后package.json中出现了对应的dependencies与devDependencies:



三、Visual Studio Code安装

下载开发工具vc-code,下载地址：<https://code.visualstudio.com/>;也有解压缩版本和exe版本。还是选择压缩版本,解压到指定文件夹。进入文件找到code.exe程序,发送到桌面快捷方式,就可以通过桌面图标启动。

三、express安装

终端输入：npm install express –g 以全局方式安装express

然后输入：npm install –g express-generator; 是因为最新版本将命令行工具分离出来了。

然后终端输入：express –version,查看版本