     Vue.js（读音 /vjuː/, 类似于 view）是一个构建数据驱动的 web 界面的渐进式框架。Vue.js 的目标是通过尽可能简单的 API 实现响应的数据绑定和组合的视图组件。它不仅易于上手，还便于与第三方库或既有项目整合。

       下面介绍三种 Vue.js 的安装方法：

        1.独立版本

       我们可以在Vue.js的官网上直接下载vue.js,并在<script>标签中引用。->  <script src = ../vue.js> </script>  开发环境不要使用最小压缩版，不然会没有错误提示和警告！

        2.使用CDN方法

 BootCDN（国内） : https://cdn.bootcss.com/vue/2.2.2/vue.min.js ， （国内不稳定）

unpkg：https://unpkg.com/vue/dist/vue.js, 会保持和 npm 发布的最新的版本一致。（推荐使用）

cdnjs : https://cdnjs.cloudflare.com/ajax/libs/vue/2.1.8/vue.min.js，如（<script src="https://cdnjs.cloudflare.com/ajax/libs/vue/2.1.8/vue.min.js"></script>）

      3.NPM方法

      在用Vue.js构建大型应用的时候推荐使用NPM安装方法，NPM能很好的和诸如Webpack或者Browserify 模块打包器配合使用。Vue.js 也提供配套工具来开发单文件组件。

     首先，先列出我们接下来需要的东西：

node.js环境（npm包管理器）

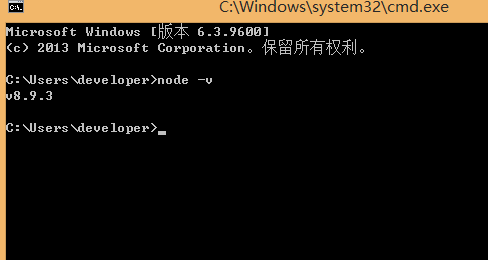
vue-cli 脚手架构建工具

cnpm npm的淘宝镜像

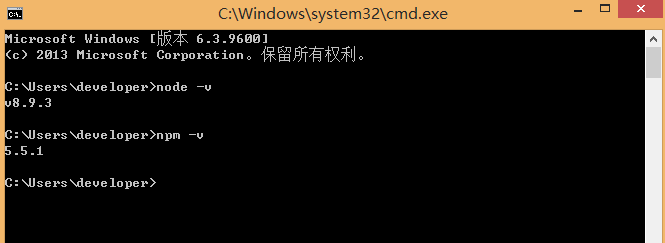
       1） 安装node.js

            从node.js官网下载并安装node，安装过程很简单，一直点下一步就ok了，安装完之后，我们通过打开命令行工具（win+R）,输入

        node -v 命令，查看node的版本，若出现相应的版本号，则说明你安装成功了。



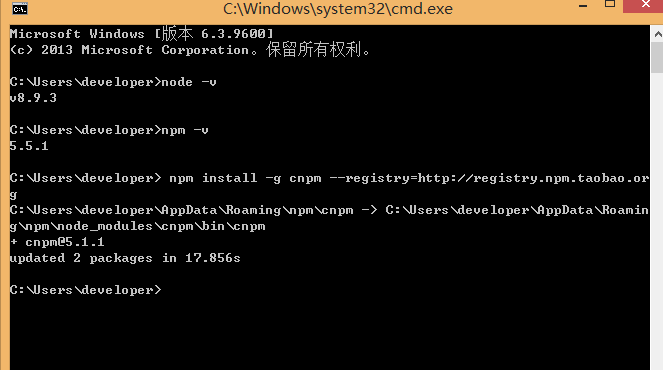
             npm包管理器，是集成在node中的，所以安装了node也就有了npm,直接输入 npm -v 命令，显示npm的版本信息。



        到目前为止，node的环境已经安装完成，npm 包管理器也有了，由于有些npm资源被屏蔽或者是国外资源的原因，经常会导致npm安装依赖包的时候失败，所以我们还需要npm的国内镜像----cnpm.

     2） 安装cnpm

           在命令行中输入  npm install -g cnpm --registry=http://registry.npm.taobao.org  ，然后等待，没报错表示安装成功，（我的已经安装过了，显示更新成功的信息），如下图：



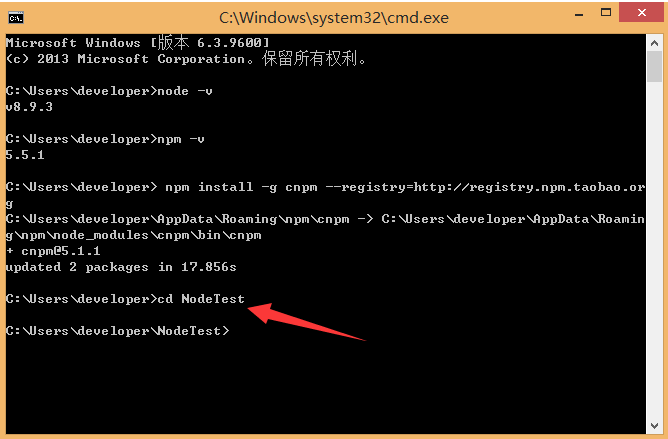
       完成之后，我们就可以用cnpm代替npm来安装依赖包了。如果想进一步了解cnpm的，查看淘宝npm镜像官网。

      3） 安装vue-cli 脚手架构建工具

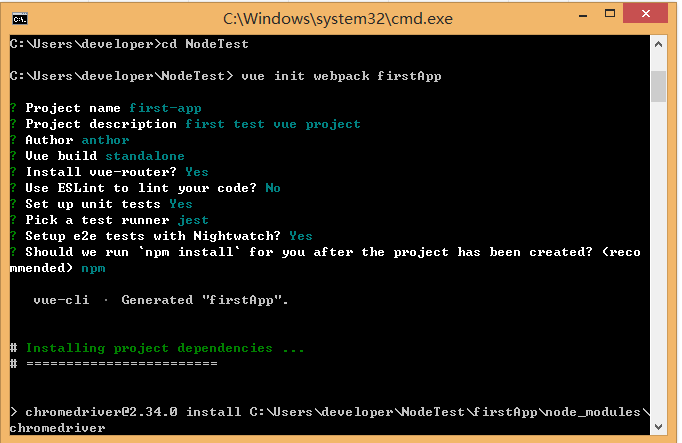
在命令行中运行命令 npm install -g vue-cli ，然后等待安装完成。

通过以上三部，我们需要准备的环境和工具都准备好了，接下来就开始使用vue-cli来构建项目。

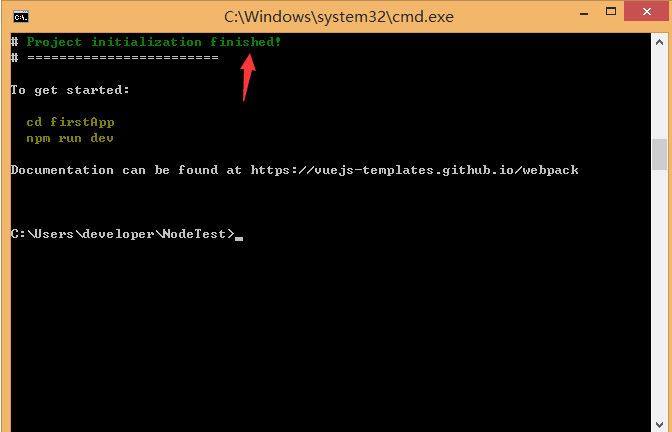
首先我们要选择存放项目的位置，然后再用命令行将目录转到选定的目录，在这里，我选择在c盘下创建新的目录（NodeTest 目录），用cd 将目录切到该目录下，如下图：



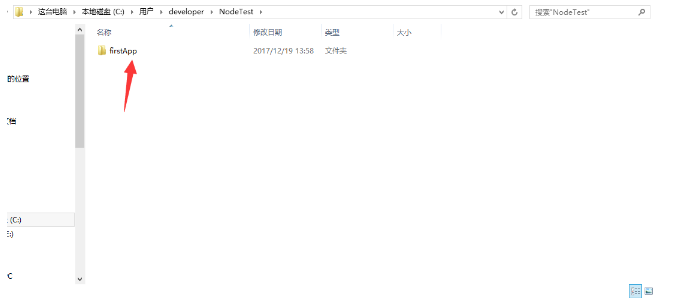
在NodeTest 目录下，在命令行中运行命令 vue init webpack firstApp 。解释一下这个命令，这个命令的意思是初始化一个项目，其中webpack是构建工具，也就是整个项目是基于webpack的。其中firstApp是整个项目文件夹的名称，这个文件夹会自动生成在你指定的目录中（我的实例中，会在NodeTest 目录生成该文件夹），如下图。



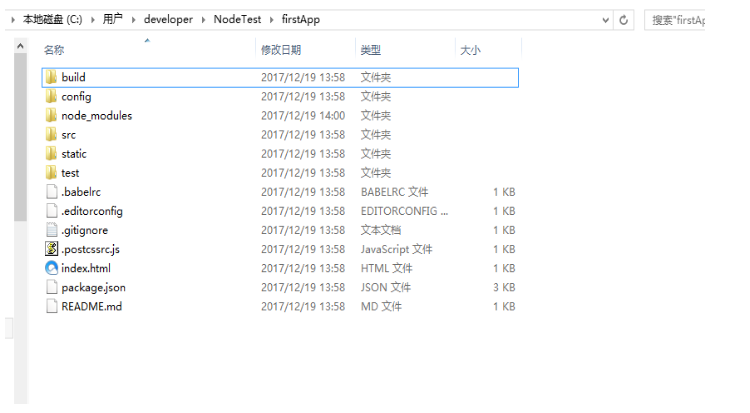
运行初始化命令的时候会让用户输入几个基本的配置选项，如项目名称、项目描述、作者信息，对于有些不明白或者不想填的信息可以一直按回车去填写就好了，等待一会，就会显示创建项目创建成功，如下图:



 接下来，我们去NoteTest目录下去看是否已创建文件：



  打开firstApp 项目，项目中的目录如下：



    介绍一下目录及其作用：

     build：最终发布的代码的存放位置。

     config：配置路径、端口号等一些信息，我们刚开始学习的时候选择默认配置。

     node\_modules：npm 加载的项目依赖模块。

     src：这里是我们开发的主要目录，基本上要做的事情都在这个目录里面，里面包含了几个目录及文件：

             assets:放置一些图片，如logo等

             components:目录里放的是一个组件文件，可以不用。

             App.vue：项目入口文件，我们也可以将组件写这里，而不使用components目录。

            main.js ：项目的核心文件

     static：静态资源目录，如图片、字体等。

     test：初始测试目录，可删除

      .XXXX文件：配置文件。

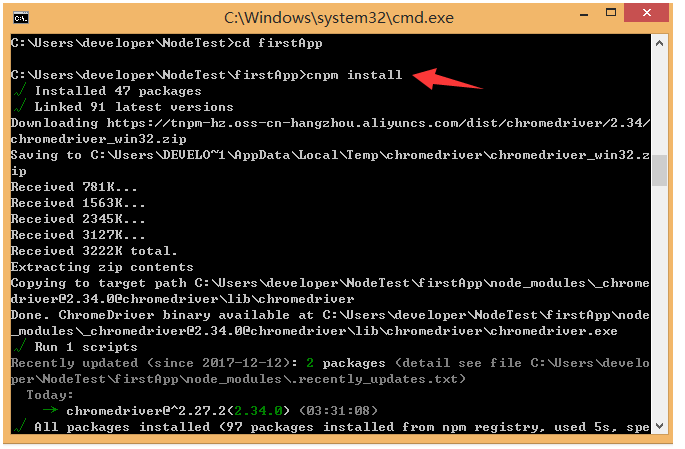
     index.html：首页入口文件，可以添加一些meta信息或者同统计代码啥的。

     package.json：项目配置文件。

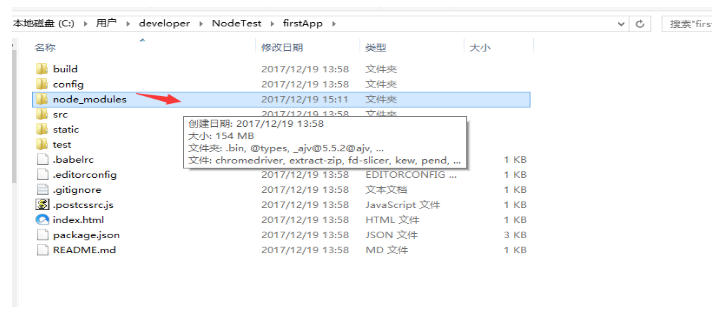
     README.md：项目的说明文件。

    这就是整个项目的目录结构，其中，我们主要在src目录中做修改。这个项目现在还只是一个结构框架，整个项目需要的依赖资源都还没有安装。

    安装项目所需要的依赖：执行 cnpm install   (这里可以用cnpm代替npm了)



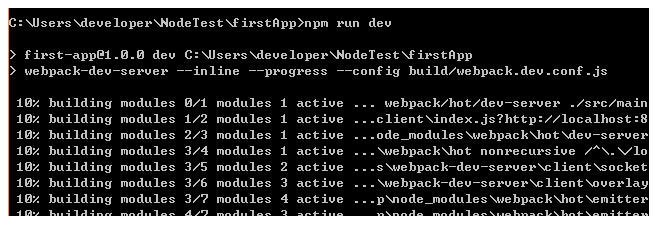
    安装完成之后，我们到自己的项目中去看，会多一个node\_modules文件夹，这里面就是我们所需要的依赖包资源。



    安装完依赖包资源后，我们就可以运行整个项目了。

   运行项目

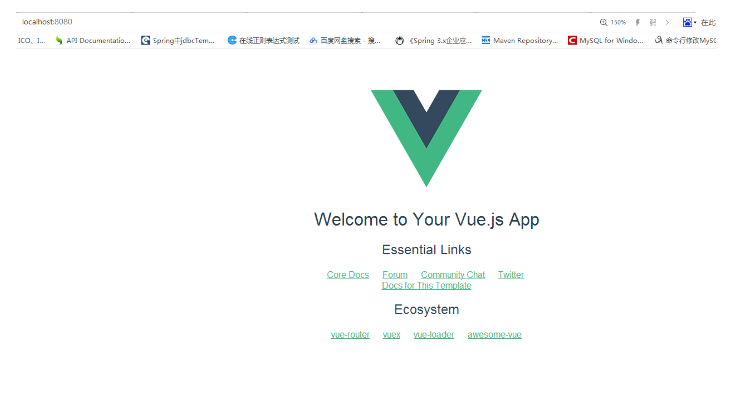
    在项目目录中，运行命令 npm run dev ，会用热加载的方式运行我们的应用，热加载可以让我们在修改完代码后不用手动刷新浏览器就能实时看到修改后的效果。



    项目启动后，在浏览器中输入项目启动后的地址：



    在浏览器中会出现vue的logo：



    至此，vue的三种安装方式已介绍完毕

---------------------

作者：程序媛-jjl

来源：CSDN

原文：https://blog.csdn.net/m0\_37479246/article/details/78836686

版权声明：本文为博主原创文章，转载请附上博文链接！