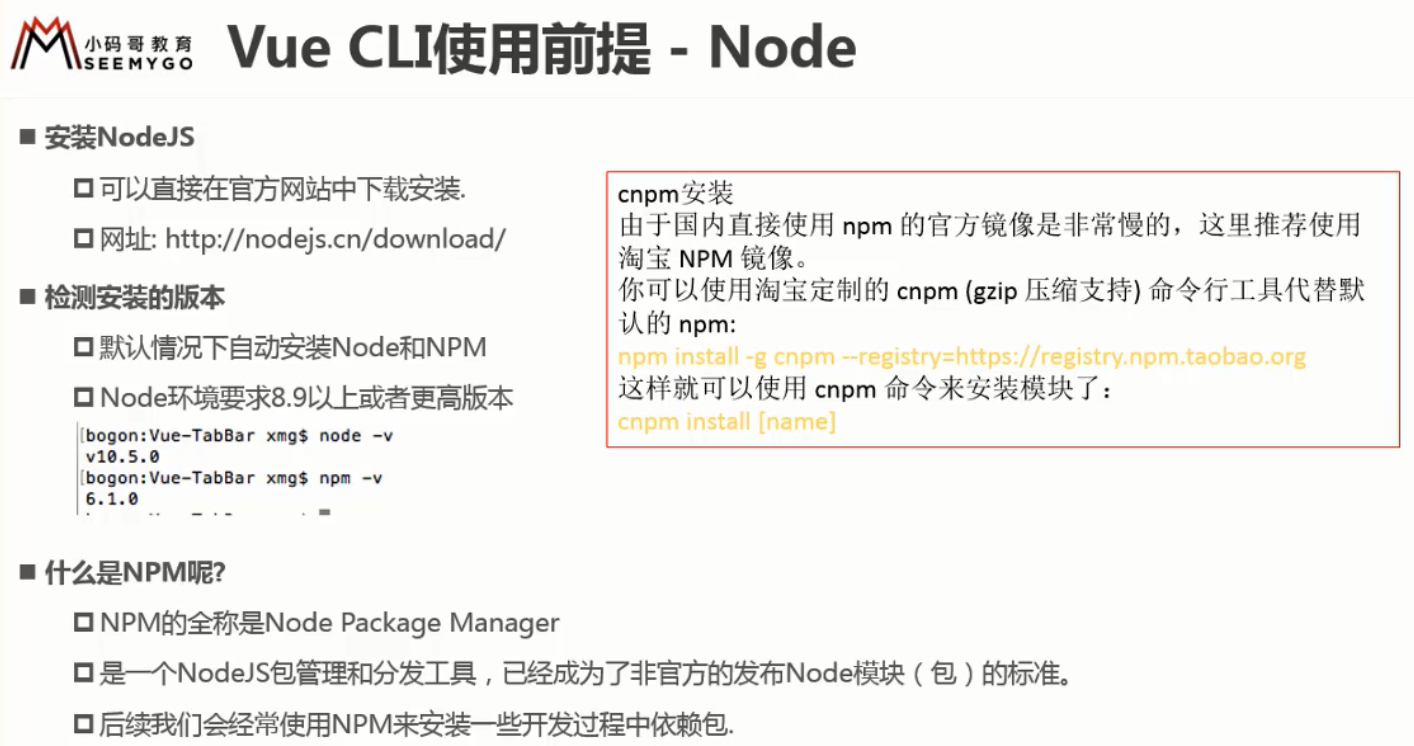
# 脚手架

## 脚手架介绍和安装



* Vue cli 使用前提
  + Node



检查vue-cli是否安装成功：命令【 vue -V 】

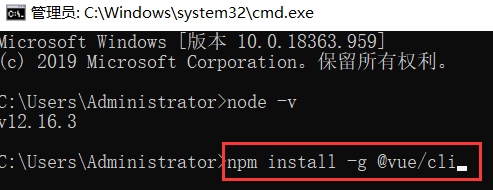
* + Webpack

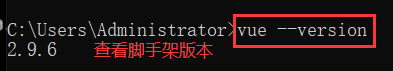


* Vue CLi的使用

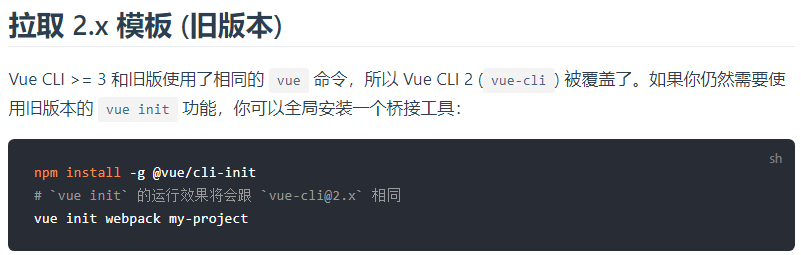


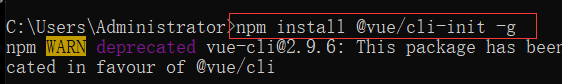
* + 直接安装脚手架3，在脚手架3的基础上，拉一个脚手架2的模板，就可以使用脚手架2。这样就能同时使用脚手架2和脚手架3了。



安装好后查看版本

* + 拉取脚手架2的模板：



拉取脚手架2模板的命令

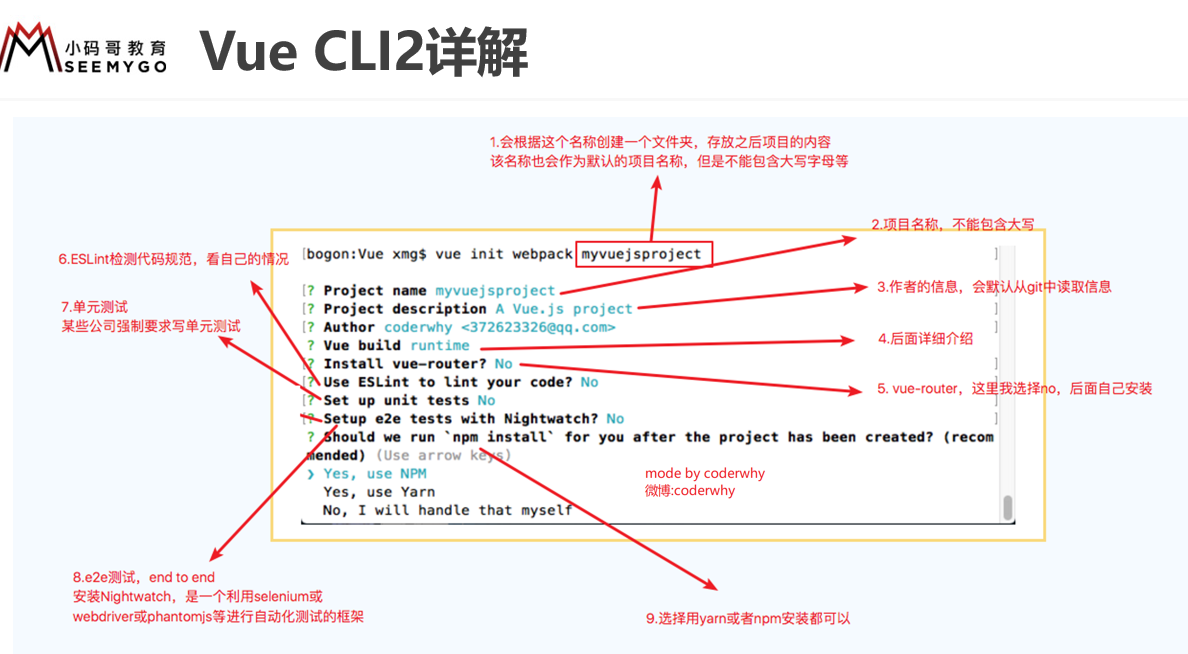
* + 创建VueCLi 2项目

## Vuecli- CLi2初始化项目过程

1. 创建脚手架2项目命令：vue init webpack vuecli2test

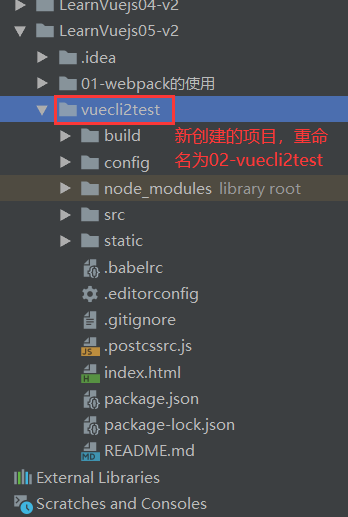


1. Vuecli2详解



选择runtime-only是因为打包的文件小，速率高，ESlint（对JS代码的限制，之后写的js代码会变得规范，如果写的不规范，编译器聚会报错）

但是先选择runtime-compiler，先不安装路由，



## Vuecli- CLi2的目录结构解析

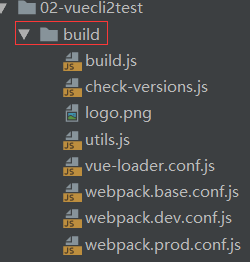
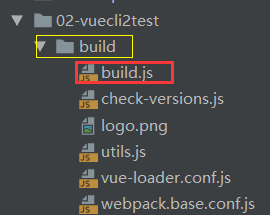
1. build 文件夹和config文件夹

都是关于webpack的配置。先从package.json文件开始读代码并理解

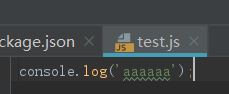
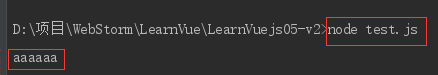


运行命令npm run dev 和npm run start一样。先找到dev和build，

1. 先看build。执行npm run build命令就是将项目打包，看build后面的代码。意思是用 node指令， 执行build文件夹下的build.js文件。node就是用来执行js文件的，node为js提供了一个运行环境。

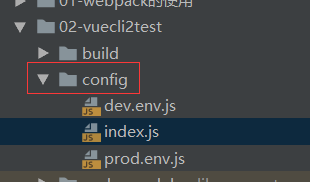
 

如新建一个test.js文件，按普通的做法还要再写一个index.html文件来引用test.js，然后再点击浏览器才能运行；但是使用node就不用那么多步骤，直接新建test.js，然后命令行输入 node test.js按回车就能看到结果，不需要再创建index.html文件，也不需要打开浏览器。

命令行

1. config文件夹。

index.js文件是用来定义一些变量的，最主要的配置在index.js文件中；另外两个是开发时配置和运行时配置。



将index.js文件中autoOpenBrowser改为true，运行后会自动打开浏览器

1. node\_modules文件夹。一般这个文件夹可以不用看。

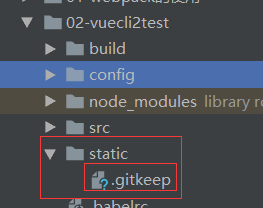
当前项目依赖的很多的包都放在该文件夹下，可以在package.json文件中查看devDependencise，里面有很多依赖的包。

1. src文件夹

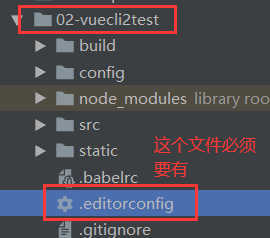
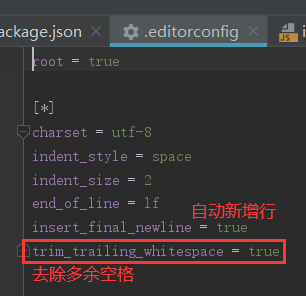
做开发的文件夹，都在这个文件夹下写代码

1. static文件夹

放一些静态的资源。打包运行后，这个文件夹下的文件会原封不动的复制到build下的dist文件夹下，不会重新命名、也不会编译成base64格式的文件，原来文件夹下是怎样的命名和格式，打包后还是原来的命名和格式。

.gitkeep文件保证无论static文件夹是否为空，都会进行上传

1. .editorconfig文件

1. .gitignore文件。不需要上传的写在这个文件中，如.idea文件夹打包时不需要上传。
2. 根目录下的index.html，这是一个HTML模板，打包上传时就根据这个文件模板进行上传
3. package.json文件，管理一些包的东西和信息管理

## 知识回顾