

## 一、Node的安装

### 1.1-安装包的方式(不推荐)

### 1.2 NVM 的方式安装

1.2.1 下载: <https://github.com/coreybutler/nvm-windows/releases>,

1.2.2 安装之前的准备工作

1.2.3 安装

1.2.4 检测 (检查是否安装成功)

1.2.5 升级

1.2.6 nvm的使用

### 1.3 nodejs的安装使用流程

### 1.4 安装 NPM

# 一、Node的安装

## 1.1-安装包的方式(不推荐)

安装方式一-----安装包的方式（不推荐）

安装包的方式（不推荐）

- Darwin: 链接
- Windows: x64 / x86

安装过程:

- Next
- Next
- Next

更新 Node 版本

- 重新下载最新的安装包;

问题: 覆盖安装即可

- 以前版本安装的很多全局的工具包需要重新安装
- 无法回滚到之前的版本
- 无法在多个版本之间切换 (很多时候我们要使用特定版本)

## 1.2 NVM 的方式安装

安装方式二

nvm(node版本管理工具,可以运行在多种操作系统上)

Windows 下的 nvm-windows 工具:

### 1.2.1 下载:

<https://github.com/coreybutler/nvm-windows/releases>,

- 这里有四个可下载的文件:
  - nvm-noinstall.zip: 绿色免安装版本,但是使用之前需要配置
  - nvm-setup.zip: 这是一个安装包,下载之后点击安装,无需配置就可以使用
  - Source code(zip): zip压缩的源码
  - Source code(tar.gz): tar.gz的源码,一般用于\*nix系统

### 1.2.2 安装之前的准备工作

**注意:** 在安装nvm for windows之前,你需要卸载任何现有版本的node.js。并且需要删除现有的nodejs安装目录(例如:"C:\Program Files\nodejs")。因为,nvm生成的symlink(符号链接/超链接)不会覆盖现有的(甚至是空的)安装目录。

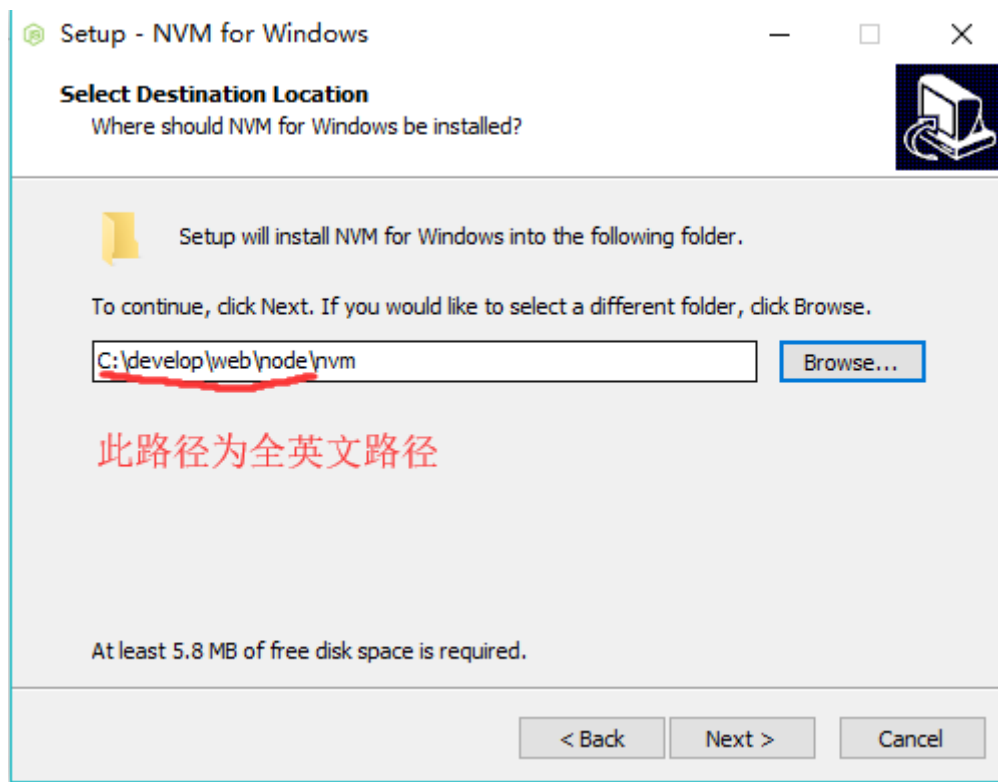
你还需要删除现有的npm安装位置(例如"C:\Users\weiqin\AppData\Roaming\npm"),以便正确使用nvm安装位置。

### 1.2.3 安装

这里将两种安装方式

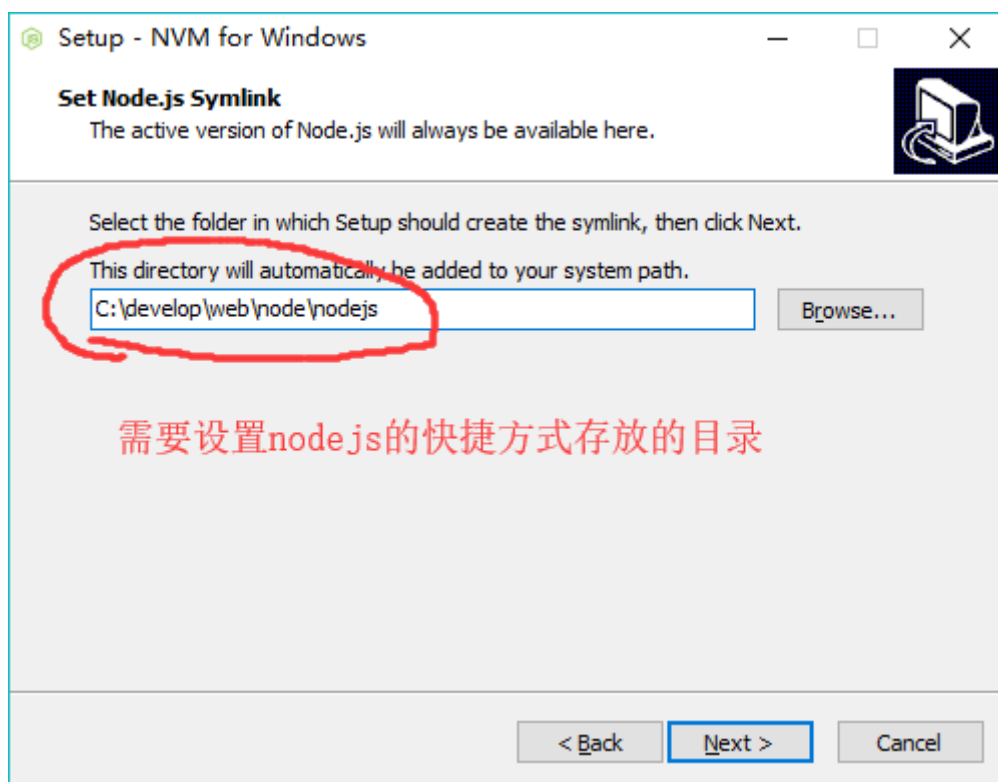
第一种: 安装包安装 nvm-setup.zip

- 1.解压 nvm-setup.zip得到 nvm-setup.exe文件
- 2.执行 nvm-setup.exe
- 3.选择安装路径



此路径为全英文路径

4.next之后,(这个目录是设置nodejs的快捷方式存放的目录)

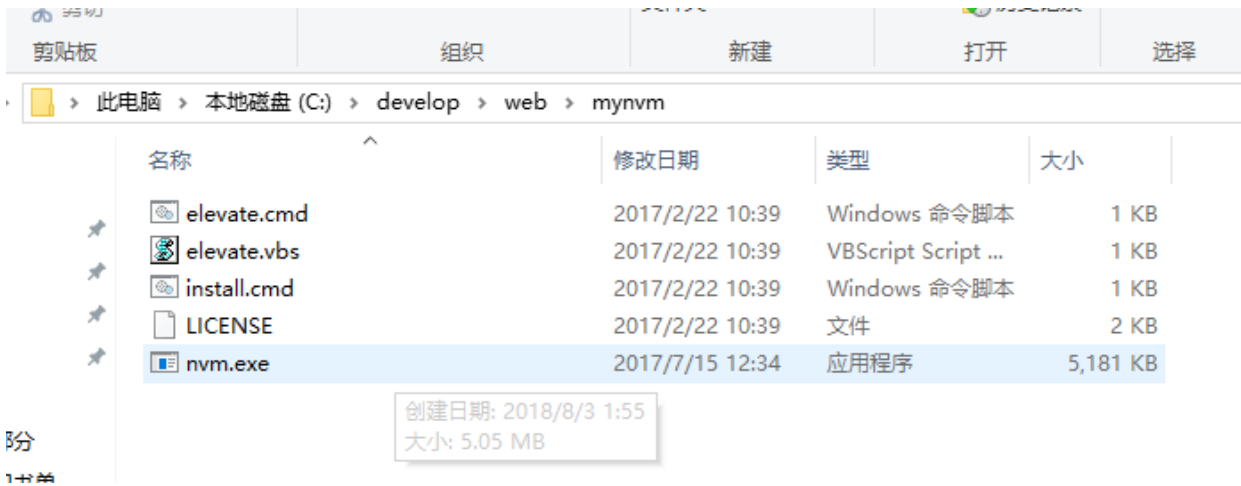


需要设置nodejs的快捷方式存放的目录

5.Next->Install->Finish完成本次安装。

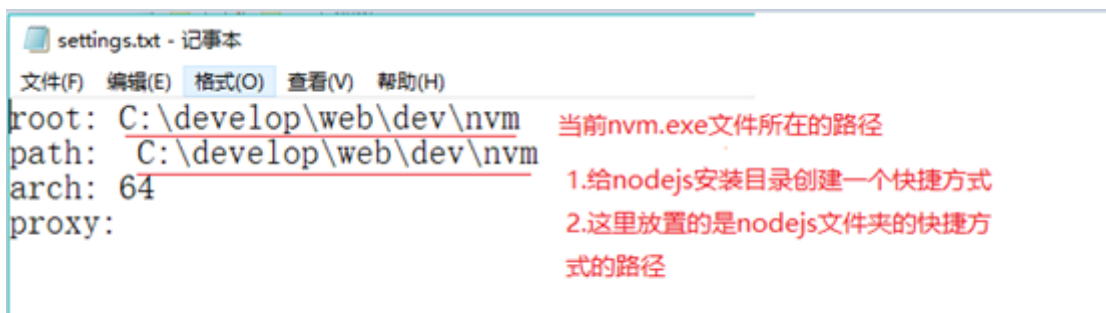
第二种：绿色安装 nvm-noinstall.zip

## 1.解压到一个全英文路径



## 2. 编辑解压目录下的 settings.txt 文件 (不存在则新建)

- root 配置为当前 nvm.exe 所在目录
- path 配置为 node 快捷方式所在的目录
- arch 配置为当前操作系统的位数 (32/64)
- proxy 不用配置



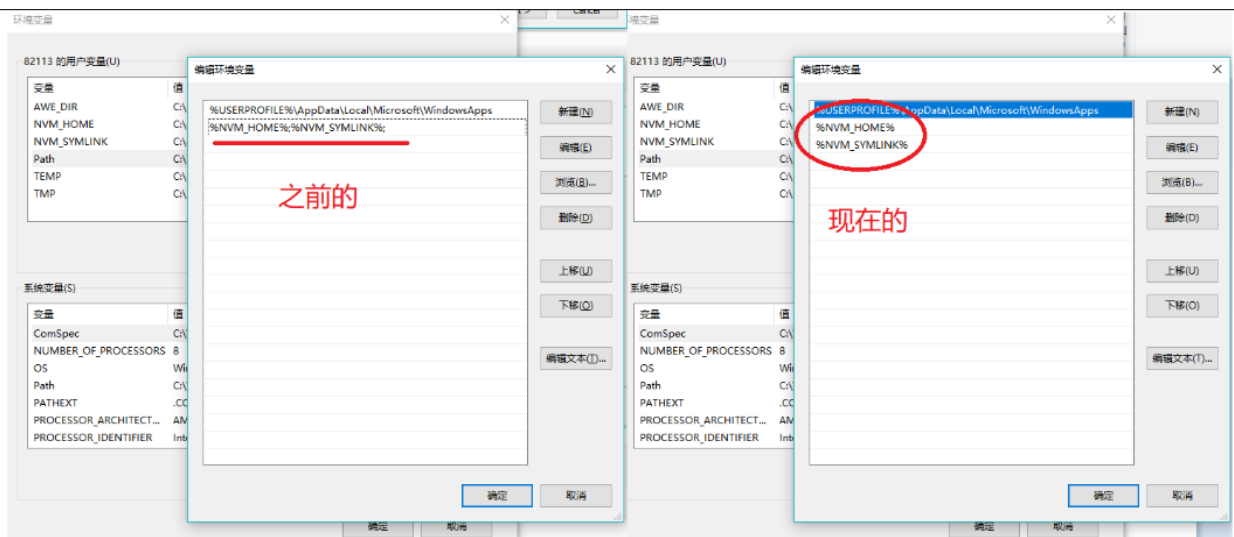
## 3.配置环境变量

注: 配置环境变量 可以通过 window+r : sysdm.cpl

NVM\_HOME = 当前 nvm.exe 所在目录

NVM\_SYMLINK = node 快捷方式所在的目录

PATH += %NVM\_HOME%;%NVM\_SYMLINK%;



打开CMD通过 `set [name]` 命令查看环境变量是否配置成功

配置失败

```
选择C:\Windows\system32\cmd.exe
Microsoft Windows [版本 10.0.16299.547]
(c) 2017 Microsoft Corporation. 保留所有权利。

C:\Users\82113>set
ALLUSERSPROFILE=C:\ProgramData
APPDATA=C:\Users\82113\AppData\Roaming
AWE_DIR=C:\Program Files (x86)\Khrona LLC\Awesomium SDK\1.6.6\
CommonProgramFiles=C:\Program Files\Common Files
CommonProgramFiles(x86)=C:\Program Files (x86)\Common Files
CommonProgramW6432=C:\Program Files\Common Files
COMPUTERNAME=DESKTOP-A9F17QF
ComSpec=C:\Windows\system32\cmd.exe
HOMEDRIVE=C:
HOMEPATH=\Users\82113
LOCALAPPDATA=C:\Users\82113\AppData\Local
LOGONSERVER=\\DESKTOP-A9F17QF
NUMBER_OF_PROCESSORS=8
NVM_HOME=C:\develop\web\node\nvm
NVM_SYMLINK=C:\develop\web\node\nodejs
OneDrive=C:\Users\82113\OneDrive
OS=Windows_NT
Path=C:\Windows\system32;C:\Windows;C:\Windows\System32\Wbem;C:\Windows\System32\WindowsPowerShell\v1.0\;C:\develop\web\node\nvm;C:\develop\web\node\nodejs;C:\Users\82113\AppData\Local\Microsoft\WindowsApps;C:\develop\web\node\nvm;C:\develop\web\node\nodejs;
PATHEXT=.COM;.EXE;.BAT;.CMD;.VBS;.VBE;.JS;.JSE;.WSF;.WSH;.MSC
PROCESSOR_ARCHITECTURE=AMD64
PROCESSOR_IDENTIFIER=Intel64 Family 6 Model 94 Stepping 3, GenuineIntel
PROCESSOR_LEVEL=6
PROCESSOR_REVISION=5e03
ProgramData=C:\ProgramData
```

配置成功

```
C:\Windows\system32\cmd.exe
No installations recognized.

C:\Users\82113>set
ALLUSERSPROFILE=C:\ProgramData
APPDATA=C:\Users\82113\AppData\Roaming
AWE_DIR=C:\Program Files (x86)\Khrona LLC\Awesomium SDK\1.6.6\
CommonProgramFiles=C:\Program Files\Common Files
CommonProgramFiles(x86)=C:\Program Files (x86)\Common Files
CommonProgramW6432=C:\Program Files\Common Files
COMPUTERNAME=BIGSPINACH
ComSpec=C:\Windows\system32\cmd.exe
HOMEDRIVE=C:
HOMEPATH=\Users\82113
LOCALAPPDATA=C:\Users\82113\AppData\Local
LOGONSERVER=\\BIGSPINACH
NUMBER_OF_PROCESSORS=8
NVM_HOME=C:\develop\web\node\nvm
NVM_SYMLINK=C:\develop\web\node\nodejs
OS=Windows_NT
Path=C:\Windows\system32;C:\Windows;C:\Windows\System32\Wbem;C:\Windows\System32\WindowsPowerShell\v1.0\;%NVM_HOME%;%NVM_SYMLINK%;C:\Users\82113\AppData\Local\Microsoft\WindowsApps;C:\develop\web\node\nvm;C:\develop\web\node\nodejs;
PATHEXT=.COM;.EXE;.BAT;.CMD;.VBS;.VBE;.JS;.JSE;.WSF;.WSH;.MSC
PROCESSOR_ARCHITECTURE=AMD64
PROCESSOR_IDENTIFIER=Intel64 Family 6 Model 94 Stepping 3, GenuineIntel
PROCESSOR_LEVEL=6
PROCESSOR_REVISION=5e03
ProgramData=C:\ProgramData
ProgramFiles=C:\Program Files
ProgramFiles(x86)=C:\Program Files (x86)
ProgramW6432=C:\Program Files
```

没有''说明配置正确

PowerShell中是通过 `dir env:[name]` 命令

NVM使用说明: <https://github.com/coreybutler/nvm-windows/>

## 1.2.4 检测 (检查是否安装成功)

命令行输入nvm查看

```
C:\Windows\system32\cmd.exe
Microsoft Windows [版本 10.0.16299.547]
(c) 2017 Microsoft Corporation. 保留所有权利。

C:\Users\82113>nvm
Running version 1.1.6.

Usage:

  nvm arch                : Show if node is running in 32 or 64 bit mode.
  nvm install <version> [arch] : The version can be a node.js version or "latest" for the latest stable version.
                                Optionally specify whether to install the 32 or 64 bit version (defaults to system arch).
                                Set [arch] to "all" to install 32 AND 64 bit versions.
                                Add --insecure to the end of this command to bypass SSL validation of the remote download server.
  nvm list [available]    : List the node.js installations. Type "available" at the end to see what can be installed. Aliased as ls.
  nvm on                  : Enable node.js version management.
  nvm off                 : Disable node.js version management.
  nvm proxy [url]         : Set a proxy to use for downloads. Leave [url] blank to see the current proxy.
                                Set [url] to "none" to remove the proxy.
  nvm node_mirror [url]   : Set the node mirror. Defaults to https://nodejs.org/dist/. Leave [url] blank to use default url.
  nvm npm_mirror [url]    : Set the npm mirror. Defaults to https://github.com/npm/npm/archive/. Leave [url] blank to default url.
  nvm uninstall <version> : The version must be a specific version.
  nvm use [version] [arch] : Switch to use the specified version. Optionally specify 32/64bit architecture.
                                nvm use <arch> will continue using the selected version, but switch to 32/64 bit mode.
  nvm root [path]         : Set the directory where nvm should store different versions of node.js.
```

## 1.2.5 升级

如果要升级的话，请重新[下载最新的安装程序](#)。并直接运行安装程序。它将安全的覆盖需要更新的文件，而无需关心nodejs的安装。

此次安装需要确保和上次使用相同的安装目录。

如果你最初安装到默认位置，则只需一直点击“下一步”，直到完成。

## 1.2.6 nvm的使用

nvm for windows是一个命令行工具，在控制台输入nvm,就可以看到它的命令用法。基本命令有：

**nvm arch [32|64]**：显示node是运行在32位还是64位模式。指定32或64来覆盖默认体系结构。

**nvm install [arch]**：该可以是node.js版本或最新稳定版本latest。（可选[arch]）指定安装32位或64位版本（默认为系统arch）。设置[arch]为all以安装32和64位版本。在命令后面添加--insecure，可以绕过远端下载服务器的SSL验证。

**nvm list [available]**：列出已经安装的node.js版本。可选的available，显示可下载版本的部分列表。这个命令可以简写为nvm ls [available]。

**nvm on**：启用node.js版本管理。

**nvm off**：禁用node.js版本管理(不卸载任何东西)

**nvm proxy [url]**：设置用于下载的代理。留[url]空白，以查看当前的代理。设置[url]为none删除代理。

**nvm node\_mirror [url]**：设置node镜像，默认为<https://nodejs.org/dist/>。我建议设置为淘宝的镜像<https://npm.taobao.org/mirrors/node/>

**nvm npm\_mirror [url]**：设置npm镜像，默认为<https://github.com/npm/npm/archive/>。我建议设置为淘宝的镜像<https://npm.taobao.org/mirrors/npm/>

**nvm uninstall** : 卸载指定版本的nodejs。  
**nvm use [version] [arch]**: 切换到使用指定的nodejs版本。可以指定32/64位[arch]。  
**nvm use** 将继续使用所选版本, 但根据提供的值切换到32/64位模式的  
**nvm root [path]**: 设置 nvm 存储node.js不同版本的目录, 如果未设置, 将使用当前目录。  
**nvm version**: 显示当前运行的nvm版本, 可以简写为nvm -v  
  
**nvm ls-remote**: 列出全部可以安装版本号  
**nvm install 版本号**: 安装指定版本  
**nvm use 版本号**: 切换指定版本, 切换效果是全局的  
**nvm current**: 查看当前使用的版本  
**nvm ls**: 查看该系统已经安装的版本, 这个命令也能看到当前使用的是哪个版本

## 1.3 nodejs的安装使用流程

nvm ls // 查看目前已经安装的版本  
nvm install 6.10.0 // 安装指定的版本的nodejs  
nvm use 6.10.0 // 使用指定版本的nodejs  
nvm install latest//安装最新版本

```
C:\Windows\system32\cmd.exe - nvm install 6.10.0
SESSIONNAME=Console
SystemDrive=C:
SystemRoot=C:\Windows
TEMP=C:\Users\82113\AppData\Local\Temp
TMP=C:\Users\82113\AppData\Local\Temp
USERDOMAIN=BIGSPINACH
USERDOMAIN_ROAMINGPROFILE=BIGSPINACH
USERNAME=82113
USERPROFILE=C:\Users\82113
windir=C:\Windows

C:\Users\82113>node -v
'node' 不是内部或外部命令, 也不是可运行的程序 或批处理文件。 还没有安装node.js

C:\Users\82113>nvm ls
No installations recognized. 没有安装任何版本的node, js

C:\Users\82113>nvm node_mirror https://npm.taobao.org/mirrors/node/

C:\Users\82113>nvm install 6.10.0 安装指定版本的node
Downloading node.js version 6.10.0 (64-bit)...
Complete
Creating C:\develop\web\node\nvm\temp
Downloading npm version 3.10.10...
```

安装node

```

C:\Users\82113>node -v
'node' 不是内部或外部命令，也不是可运行的程序
或批处理文件。

C:\Users\82113>nvm ls
No installations recognized.

C:\Users\82113>nvm node_mirror https://npm.taobao.org/mirrors/node/

C:\Users\82113>nvm install 6.10.0
Downloading node.js version 6.10.0 (64-bit)...
Complete
Creating C:\develop\web\node\nvm\temp

Downloading npm version 3.10.10... Complete
Installing npm v3.10.10...

Installation complete. If you want to use this version, type

nvm use 6.10.0

C:\Users\82113>nvm install latest
Downloading node.js version 10.8.0 (64-bit)...
Complete
Creating C:\develop\web\node\nvm\temp

Downloading npm version 6.2.0... Download failed. Rolling Back.
Rollback failed. remove C:\develop\web\node\nvm\temp\npm-v6.2.0.zip: The process cannot access the file because it is being
used by another process.
Could not download npm for node v10.8.0.
Please visit https://github.com/npm/npm/releases/tag/v6.2.0 to download npm.
It should be extracted to C:\develop\web\node\nvm\v10.8.0

C:\Users\82113>node -v
'node' 不是内部或外部命令，也不是可运行的程序
或批处理文件。

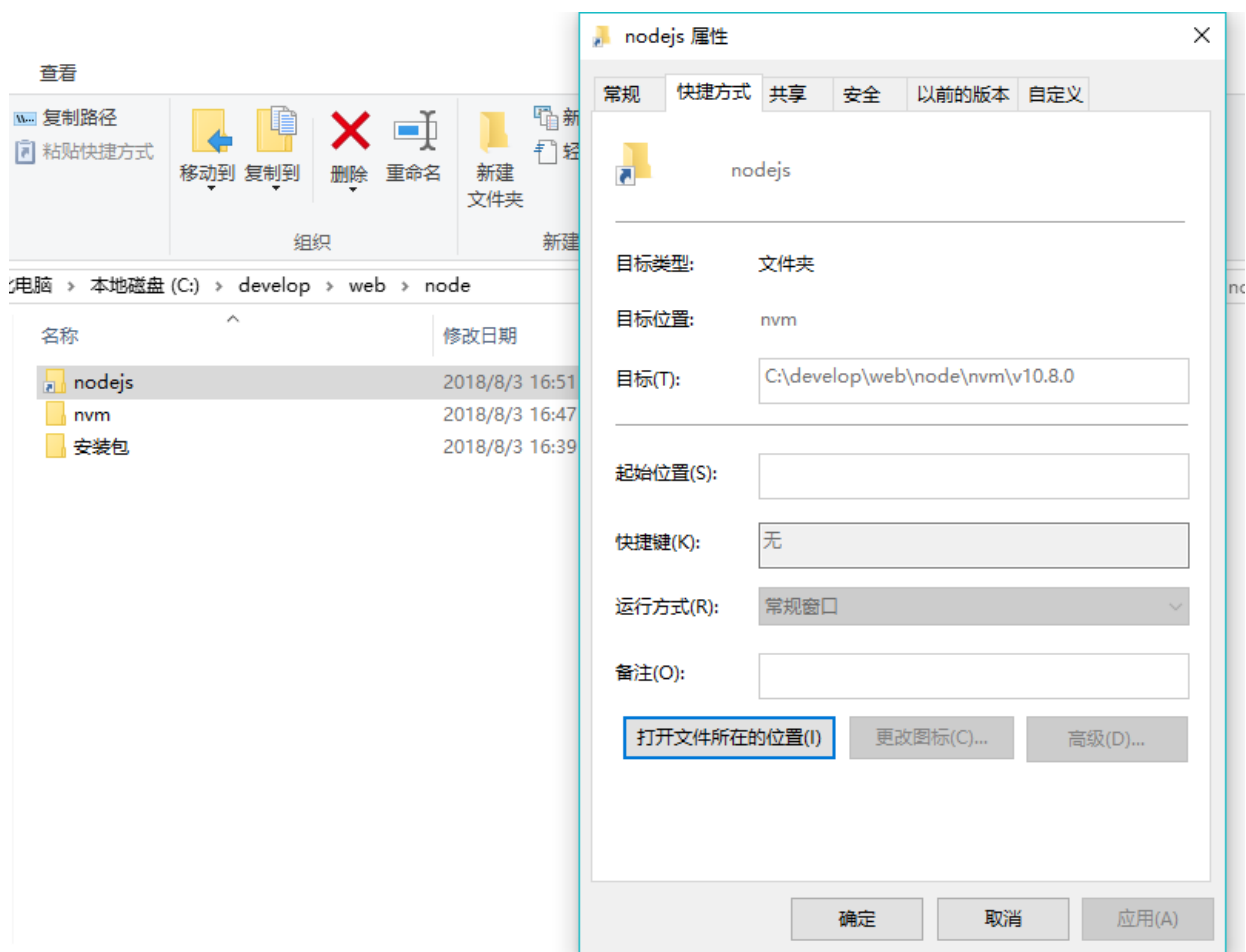
C:\Users\82113>nvm use 10.8.0
Now using node v10.8.0 (64-bit)

C:\Users\82113>node -v
v10.8.0

C:\Users\82113>_

```

本地显示效果



再次切换node版本



```
C:\Users\82113>nvm use 10.8.0
Now using node v10.8.0 (64-bit)

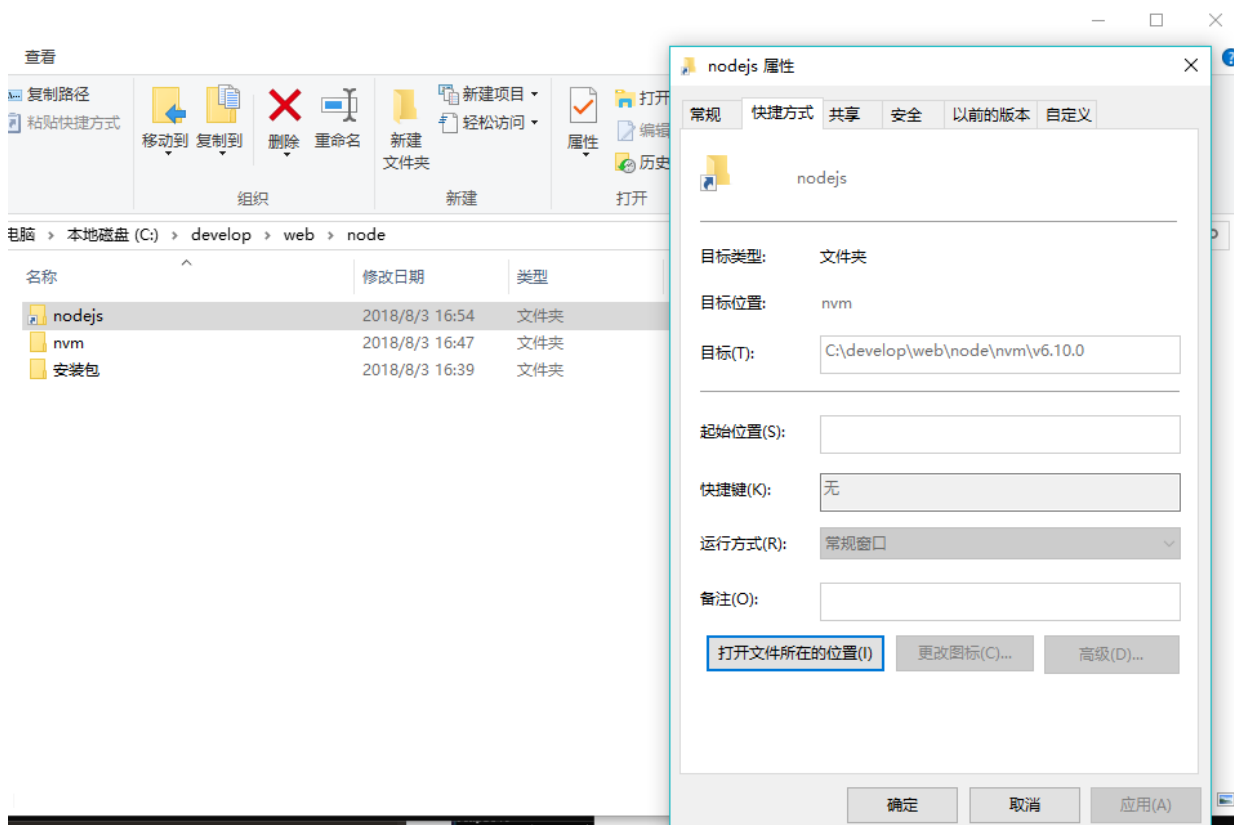
C:\Users\82113>node -v
v10.8.0

C:\Users\82113>nvm use 6.10.0
Now using node v6.10.0 (64-bit)

C:\Users\82113>node -v
v6.10.0

C:\Users\82113>
```

本地显示效果



## 1.4 安装 NPM

NPM 不需要单独安装。默认在安装 Node 的时候，会连带一起安装 NPM。但是，Node 附带的 NPM 可能不是最新版本，最好用下面的命令，更新到最新版本。

```
$ npm install npm -g
```

配置全局npm

