

VUE Cli课前准备_软件安装篇

1. 搭建开发环境

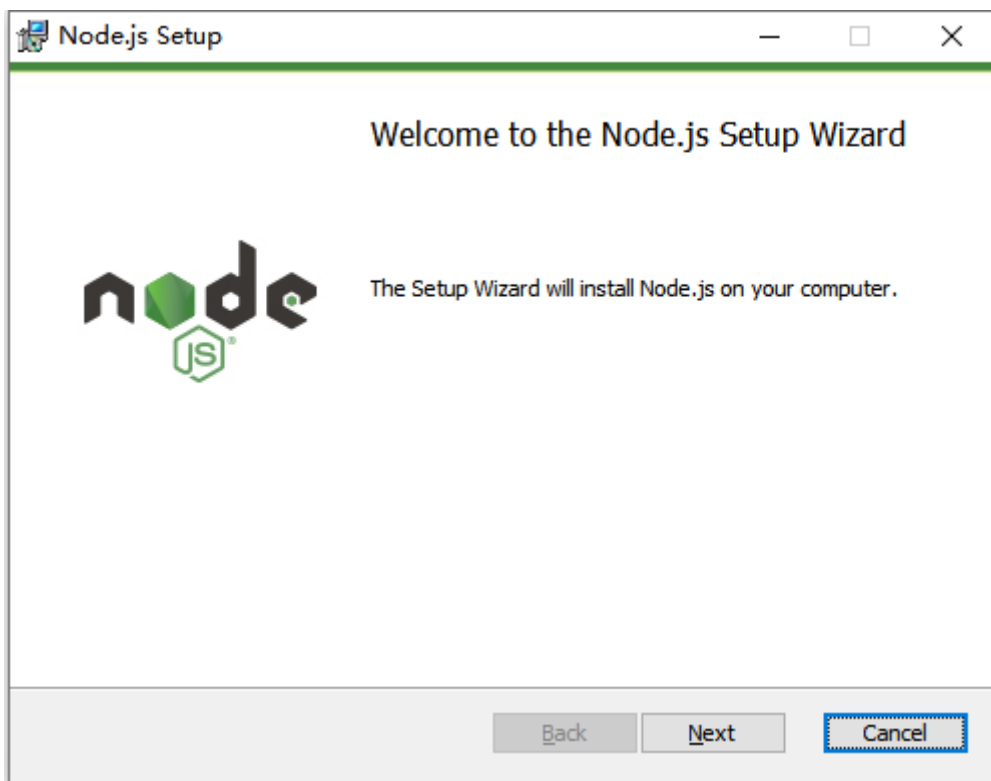
1.1. 安装Node.js

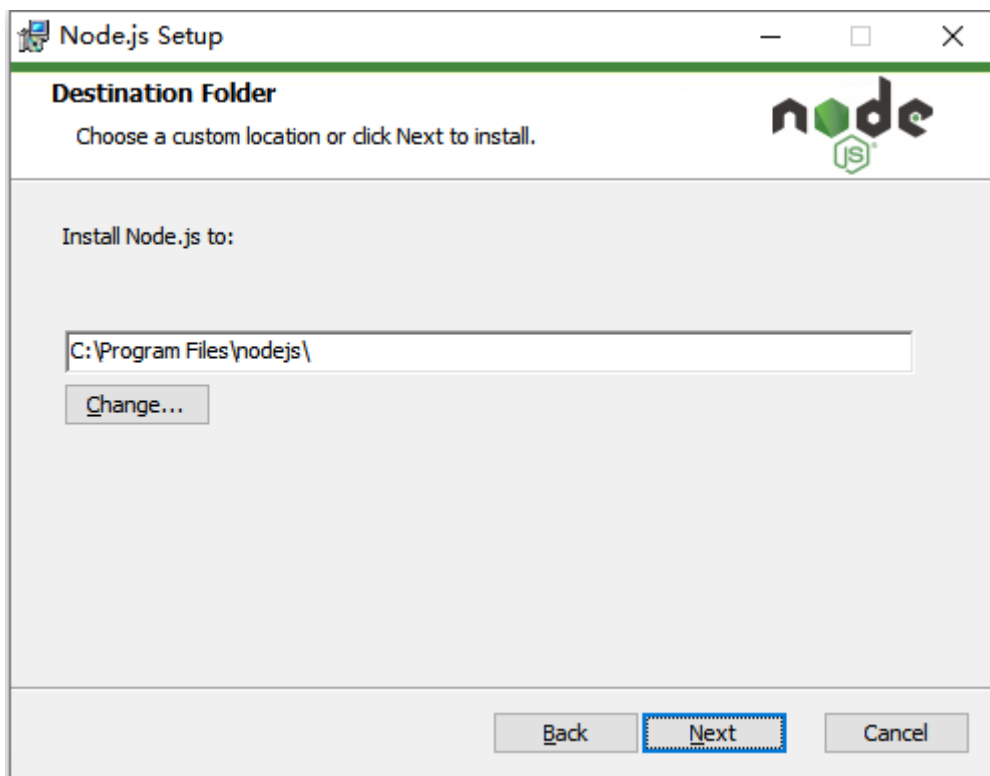
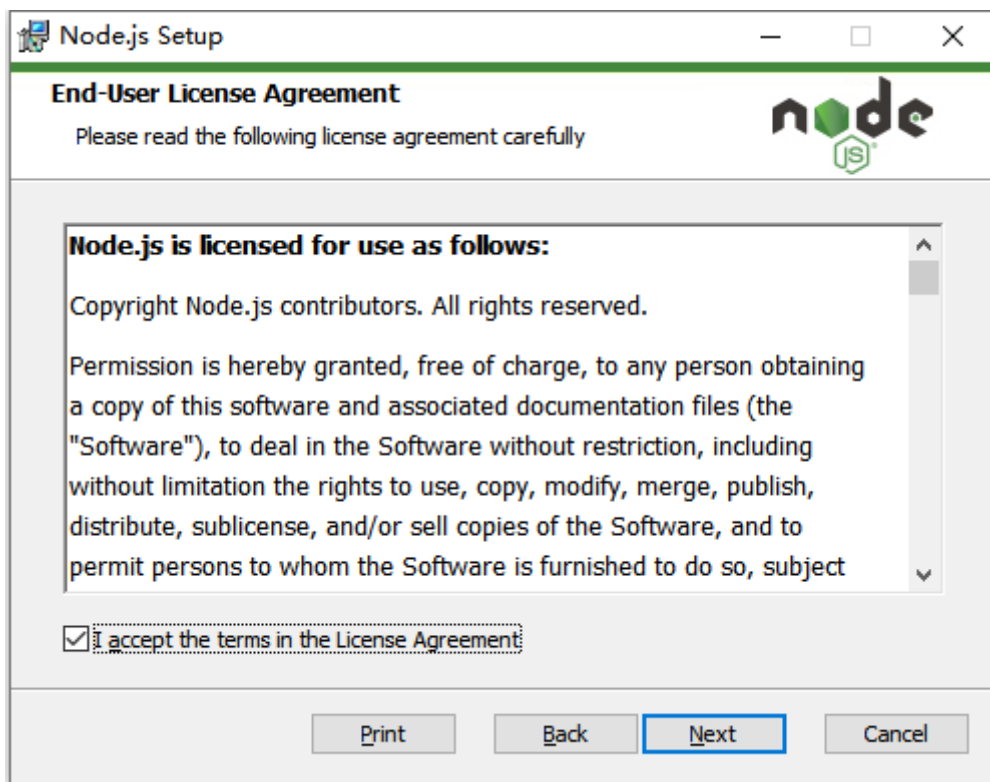
首先需要下载Node.js安装包，下载地址可参考：

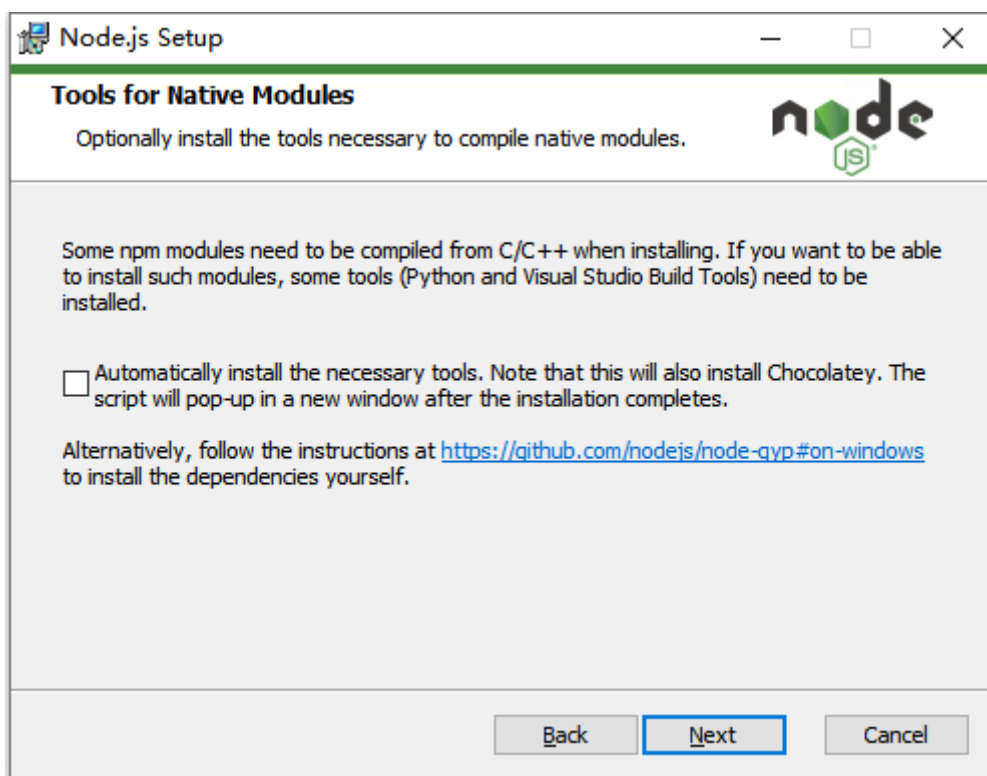
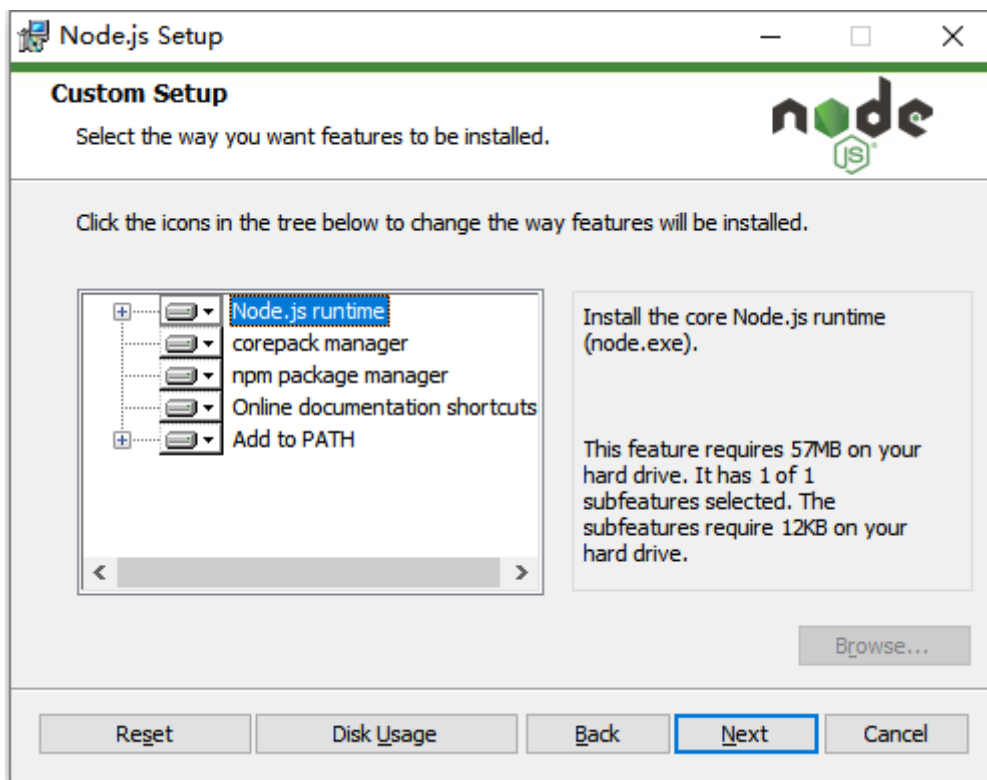
- <https://nodejs.org/dist/v16.14.2/node-v16.14.2-x64.msi>
- <https://mirrors.nju.edu.cn/nodejs/latest-v16.x/node-v16.14.2-x64.msi>
- <https://mirrors.tuna.tsinghua.edu.cn/nodejs-release/v16.14.2/node-v16.14.2-x64.msi>

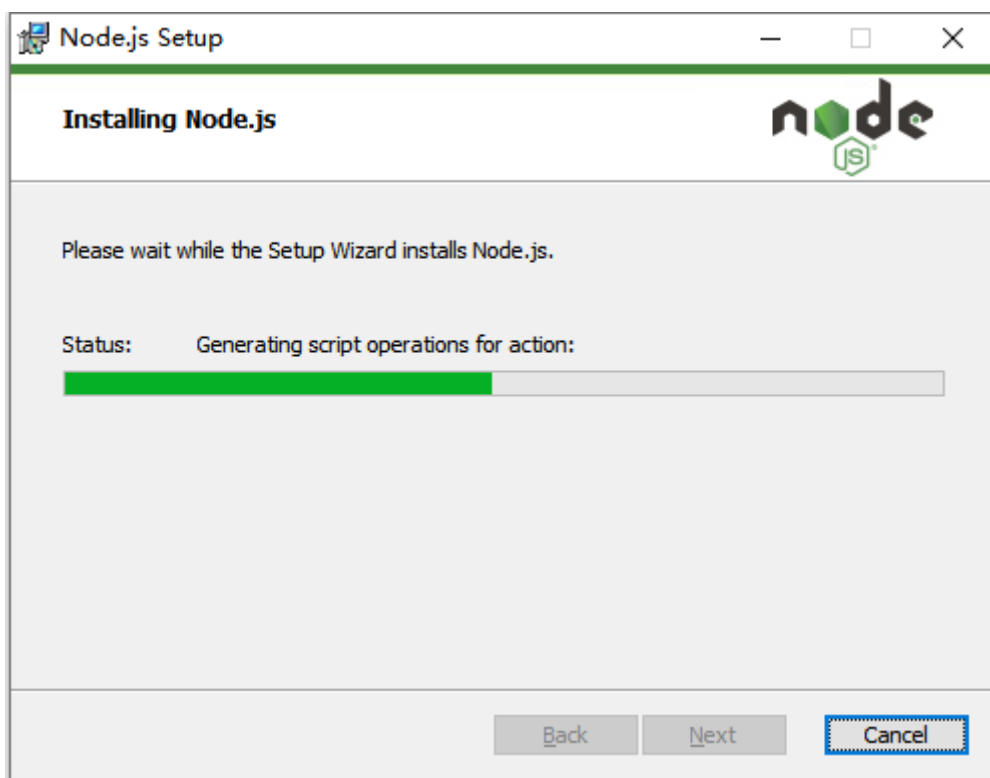
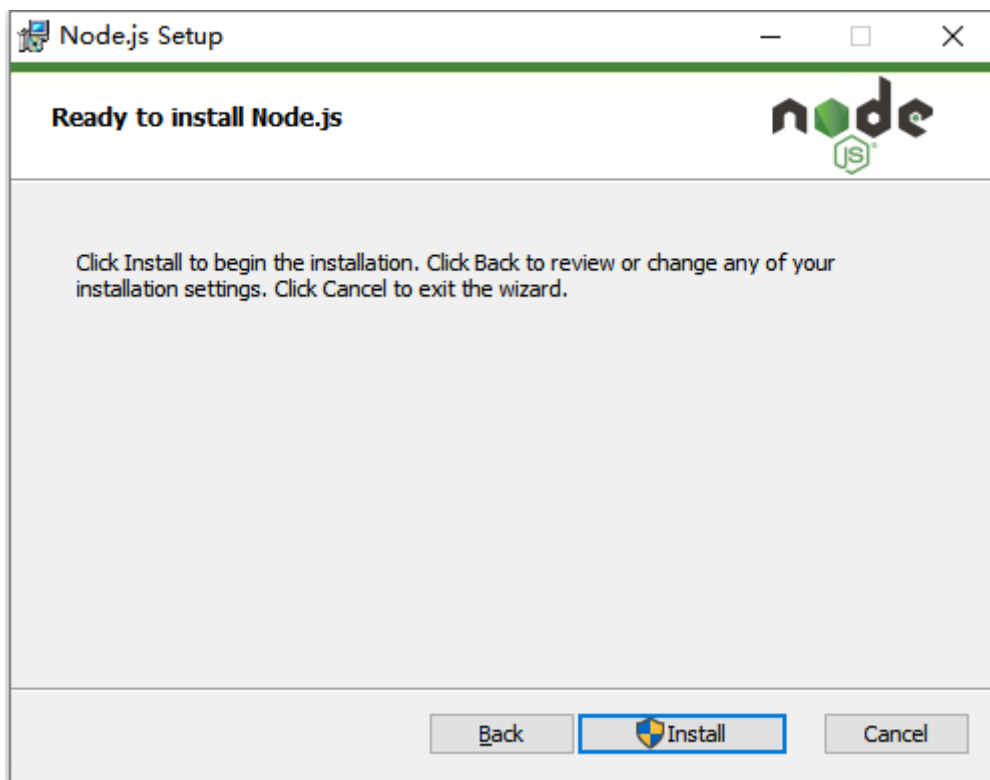
当前课程对Node.js的版本**没有严格要求**，可替换为其它稳定版本。

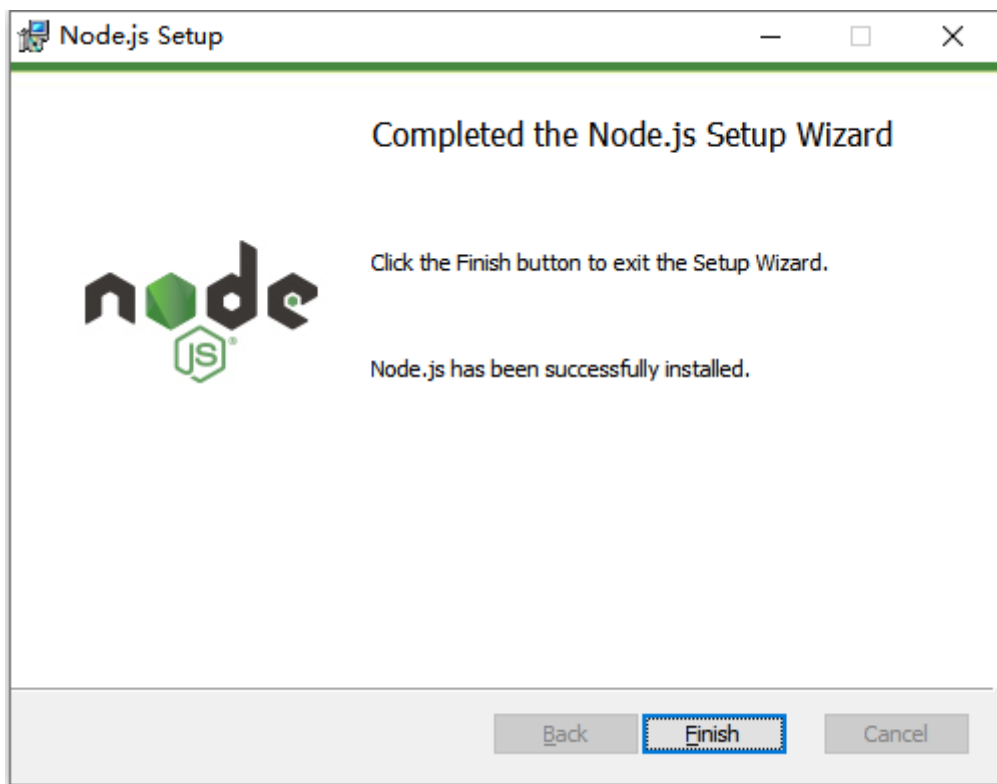
Node.js的安装过程没有特别的注意事项，整个安装过程中都可以**不必修改任何内容**，直至其自动安装完成。











此安装包会自动配置环境变量，安装完成后，可检查npm命令是否可用，在操作系统的命令提示符下执行以下命令即可：

```
npm -v
```

执行以上命令后，将显示版本号，例如：

```
C:\>npm -v  
8.5.0
```

注意：你必须在新的命令提示符窗口（安装完Node.js后再打开的窗口）中运行此命令。

1.2. 配置npm源

npm源（即npm仓库，称之为：registry）默认是境外服务器，在国内使用时，访问速度较慢，通常，在初次使用时，应该将npm源更换为国内的服务器地址，例如使用 <https://registry.npmmirror.com> 作为npm源。

配置npm源的命令如下：

```
npm config set registry https://registry.npmmirror.com
```

当配置成功后，可通过 `get` 命令可查看npm源：

```
npm config get registry
```

注意：以上命令将显示当前生效的npm源，但并不检查npm源的URL是否有效，所以，即使配置了错误的npm源，也不会提示错误，请认真检查是否配置了正确的URL。

注意：无论你使用哪种操作系统，必须保证当前登录的用户具有最高访问权限，例如，在Windows操作系统中，请使用管理员模式的命令提示符，并且，不要使用Power Shell，在Mac OS中，请使用 `sudo` 来执行以上命令。

2. 安装VUE Cli

安装VUE Cli的命令如下：

```
npm install -g @vue/cli
```

此过程将从仓库服务器中下载VUE Cli，通常耗时30秒至5分钟左右。

提示：如果安装过程中的日志提示更新npm版本，或提示某些软件版本偏旧等，只要没有出现 `ERROR` 字样，均可无视。

注意：以上操作仍需要管理权限才可以成功执行。

安装完成后，可通过以下命令检验VUE Cli是否安装成功：

```
vue -V
```

注意：以上命令中的 `-v` 中的 `v` 是大写的。

执行以上命令后，将显示当前的VUE Cli版本号，例如：

```
C:\>vue -V
@vue/cli 5.0.3
```

3. 创建第1个VUE CLI工程

3.1. 创建工程

通常使用Vue的命令来创建VUE CLI工程，并且，会将工程创建在执行命令时的位置，所以，先在命令提示符窗口中进入Vue Workspace（自行指定的用于存放项目的文件夹），例如（以下示例中的 `#` 开头是注释，不要执行）：

```
# 切换到D盘
d:

# 创建vue的workspace，文件夹名称为Vue-workspace，是自定义名称
# 如果vue的workspace已经存在，则不需要执行接下来这条命令
# 如果使用IntelliJ IDEA开发VUE CLI工程，也可以使用IntelliJ IDEA的workspace，无强制要求
mkdir Vue-workspace

# 进入D盘下的Vue-workspace
cd Vue-workspace
```

当进入Vue Workspace后，就可以创建工程了。

使用 `vue create` 命令即可创建VUE CLI工程，命令格式是：

```
vue create 工程名称
```

例如执行：

```
vue create vue-project-01
```

输入以上指令后，按下 `Enter` 键将准备创建工程，创建过程可能耗时较长，**注意：不可反复按 `Enter` 键**，否则会自动选择后续各设置选项的默认项。

创建过程中会有一些选项，最先提示的选项是 `Please pick a preset`，表示“请选择一个预设项”，推荐选择 `Manually select features`，表示“手动选择”，通过键盘的上下箭头进行选择，选择到目标项后按下键盘的 `Enter` 键到下一步：

```
Vue CLI v5.0.3
? Please pick a preset:
  Default ([Vue 3] babel, eslint)
  Default ([Vue 2] babel, eslint)
> Manually select features
```

接下来的选项是 `Check the features needed for your project`，表示“选择你的工程中需要使用功能”，推荐在列表中选择 `Babel`、`Router`、`Vuex` 这三项，使用键盘的上下箭头移动，使用空格选中或取消选中，选择完成后按下键盘的 `Enter` 键到下一步：

```
? Check the features needed for your project: (Press <space> to select, <a> to toggle all, <i> to invert selection, and <enter> to proceed)
>(*) Babel
  ( ) TypeScript
  ( ) Progressive Web App (PWA) Support
  (*) Router
  (*) Vuex
  ( ) CSS Pre-processors
  ( ) Linter / Formatter
  ( ) Unit Testing
  ( ) E2E Testing
```

接下来的选项是 `Choose a version of Vue.js that you want to start the project with`，表示“选择你的工程中希望使用的Vue.js的版本”，推荐选择 `2.x` 这项，选择到目标项后按下键盘的 `Enter` 键到下一步：

```
? Choose a version of Vue.js that you want to start the project with
  3.x
> 2.x
```

接下来的选项是 `Use history mode for router`，表示“是否在路由中选择历史模式”，推荐选择“是”，输入 `Y` 后按下键盘的 `Enter` 键到下一步（提示信息中，`Y` 是大写的，表示它是默认选项，不输入 `Y` 而直接按下 `Enter` 键是等效的）：

```
? Use history mode for router? (Requires proper server setup for index fallback in production) (Y/n) █
```

接下来的选项是 `where do you prefer placing config for Babel, ESLint, etc.?`，表示“你习惯把一些配置信息存放在哪里？”，推荐选择 `In package.json`，即存放在 `package.json` 文件中，选择到目标项后按下键盘的 `Enter` 键到下一步：

? Where do you prefer placing config for Babel, ESLint, etc.?

In dedicated config files

> In package.json

最后的选项是 `Save this as a preset for future projects?`，表示“是否保存以上配置信息，作为后续将创建的新工程的预设？”，推荐选择“否”，输入 `N` 后按下键盘的 `Enter` 键（提示信息中，`N` 是大写的，表示它是默认选项，不输入 `N` 而直接按下 `Enter` 键是等效的）：

? Save this as a preset for future projects? (y/N)

至此，配置过程已经结束，接下来会自动创建工程，耗时取决于网络速度，通常需要30秒至5分钟左右。

如果未出现 `Error` 字样，通常是创建成功了。

3.2. 启动服务

注意：从现在开始，你可以使用IntelliJ IDEA或类似的开发工具来打开你的项目，并使用内置的Terminal执行各个命令，使用方式与命令提示符窗口完全相同。

进入工程目录，执行 `npm run serve` 命令即可启动此工程的服务（过程中会编译打包此工程）：

```
npm run serve
```

当提示以下信息时，表示工程已成功启动：

```
DONE Compiled successfully in 2654ms
```

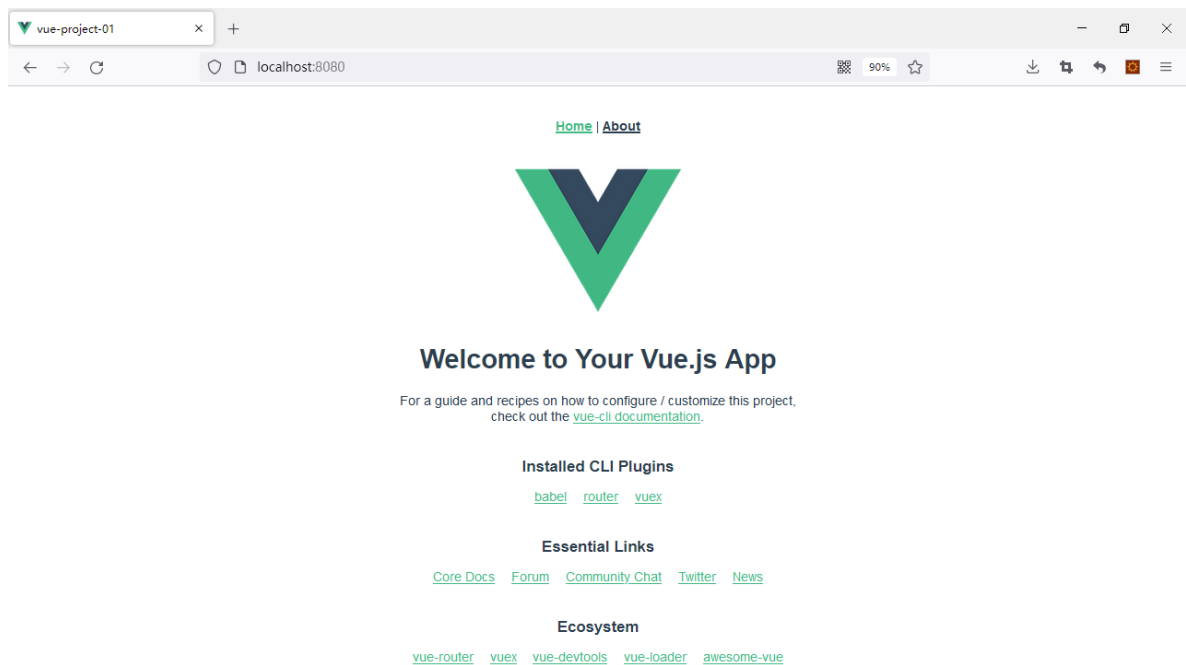
```
App running at:
```

- Local: <http://localhost:8080/>
- Network: <http://192.168.3.7:8080/>

```
Note that the development build is not optimized.
```

```
To create a production build, run npm run build.
```

当工程成功启动后，即可根据提示的URL在浏览器中进行访问：



3.3. 停止服务

如果需要停止服务，在控制台使用Ctrl + C即可。

提示：当按下Ctrl + C后，会提示 "终止批处理操作吗(Y/N)?"，其实，当按下Ctrl + C后，服务已经被停止，在此提示中无论选择Y或N结果都已经不重要了。

3.4. 重启服务

Vue脚本架工程启动的服务支持热部署（热更新，即：当保存对源代码的修改后，会自动重新编译打包并覆盖原本已经启动的服务中的文件），并没有重启服务的做法，如果在某些场景中需要实现“重启”的效果（例如修改服务端口等），可以先停止服务，然后再次启动服务。