

# Configure for Lab



**Bok, Jong Soon**  
**javaexpert@nate.com**  
**<https://github.com/swacademy/Vue.js>**

# Contents

- Web Browser
  - Google Chrome
  - Mozilla Firefox
- Editor or IDE
  - Microsoft Visual Studio Code
- Node.js
- Git
- Vue.js devtools
- Vue-CLI
- Online Editors

# Web Browsers

- Google Chrome Browser

- <https://www.google.com/chrome/>

- Mozilla Firefox

- <https://www.mozilla.org/ko/>

- Microsoft Edge

- <https://www.microsoft.com/ko-kr/edge>



# Editors or IDEs

- Visual Studio Code

- <https://code.visualstudio.com/>

- Atom

- <https://atom.io/>

- SublimeText3

- <https://www.sublimetext.com/3>




# Visual Studio Code

- <https://code.visualstudio.com/Download>

## Download Visual Studio Code

Free and built on open source. Integrated Git, debugging and extensions.




↓ Windows

Windows 7, 8, 10

User Installer64 bit32 bit

System Installer64 bit32 bit

.zip64 bit32 bit



↓ .deb


Debian, Ubuntu

.deb64 bit

.rpm64 bit

.tar.gz64 bit

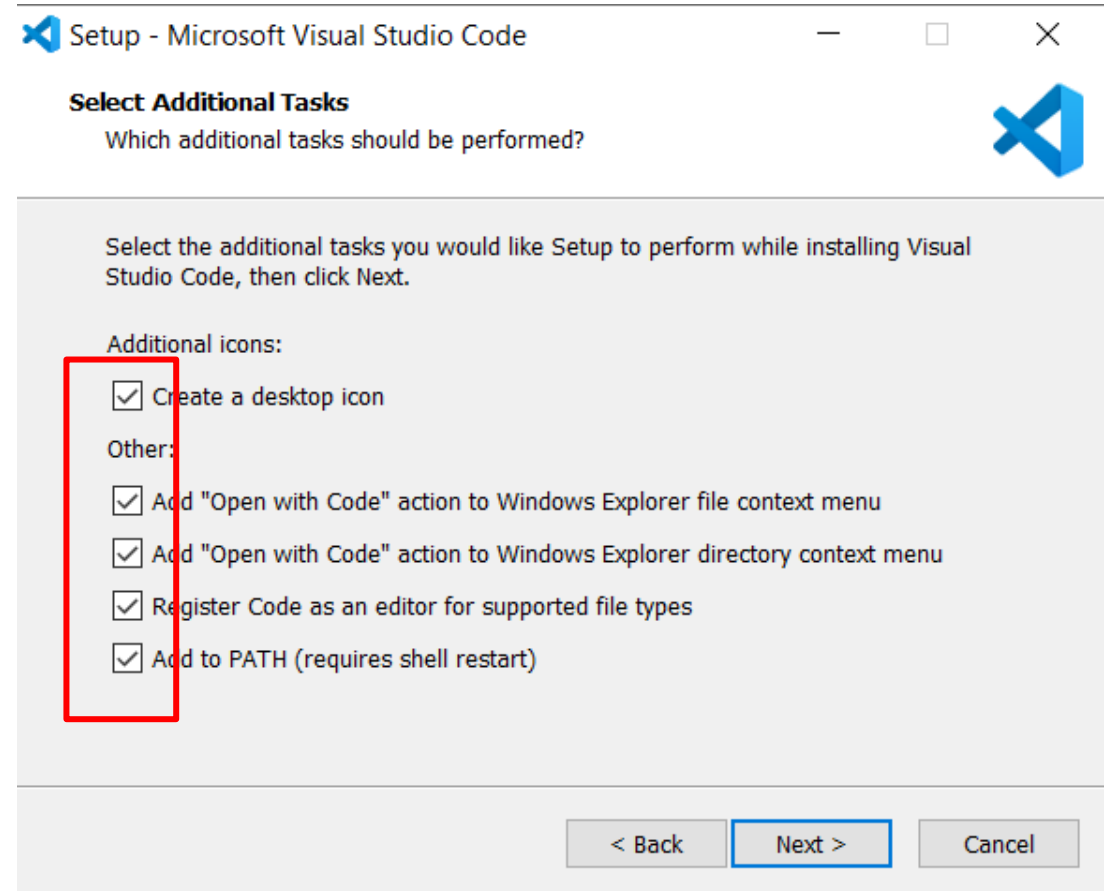
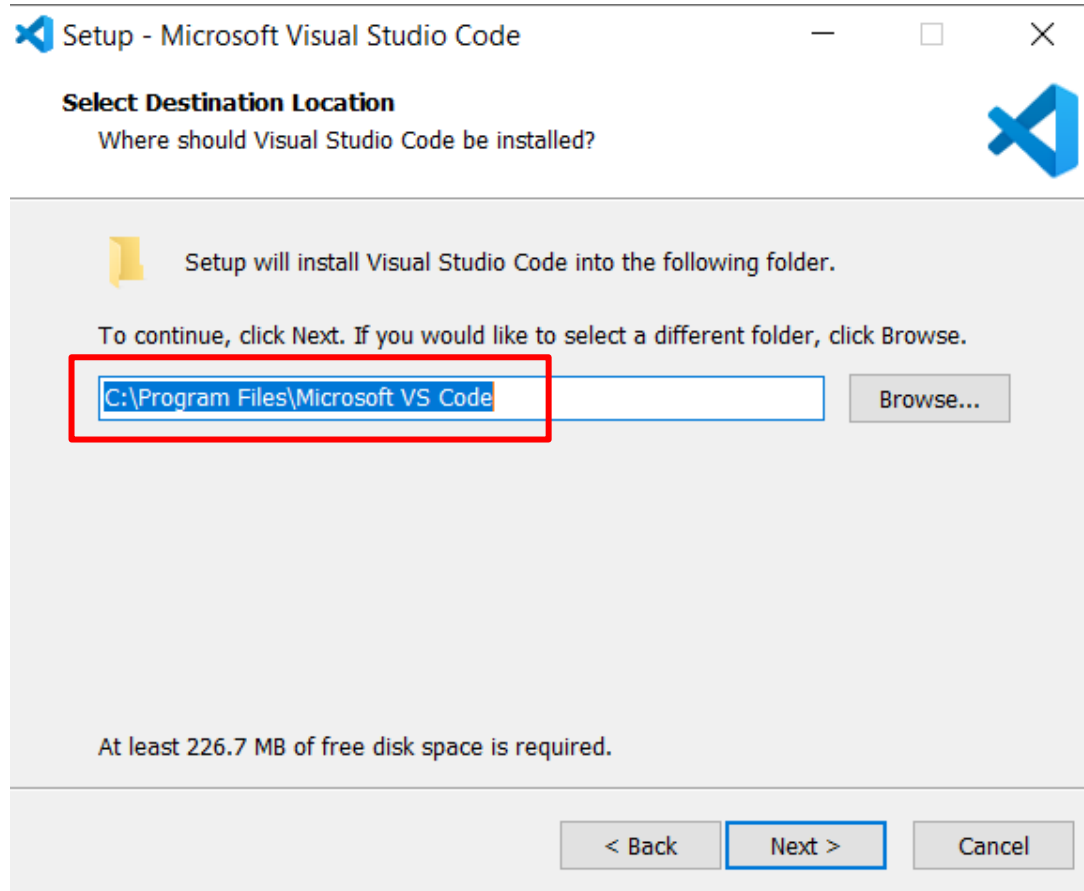
Snap Store



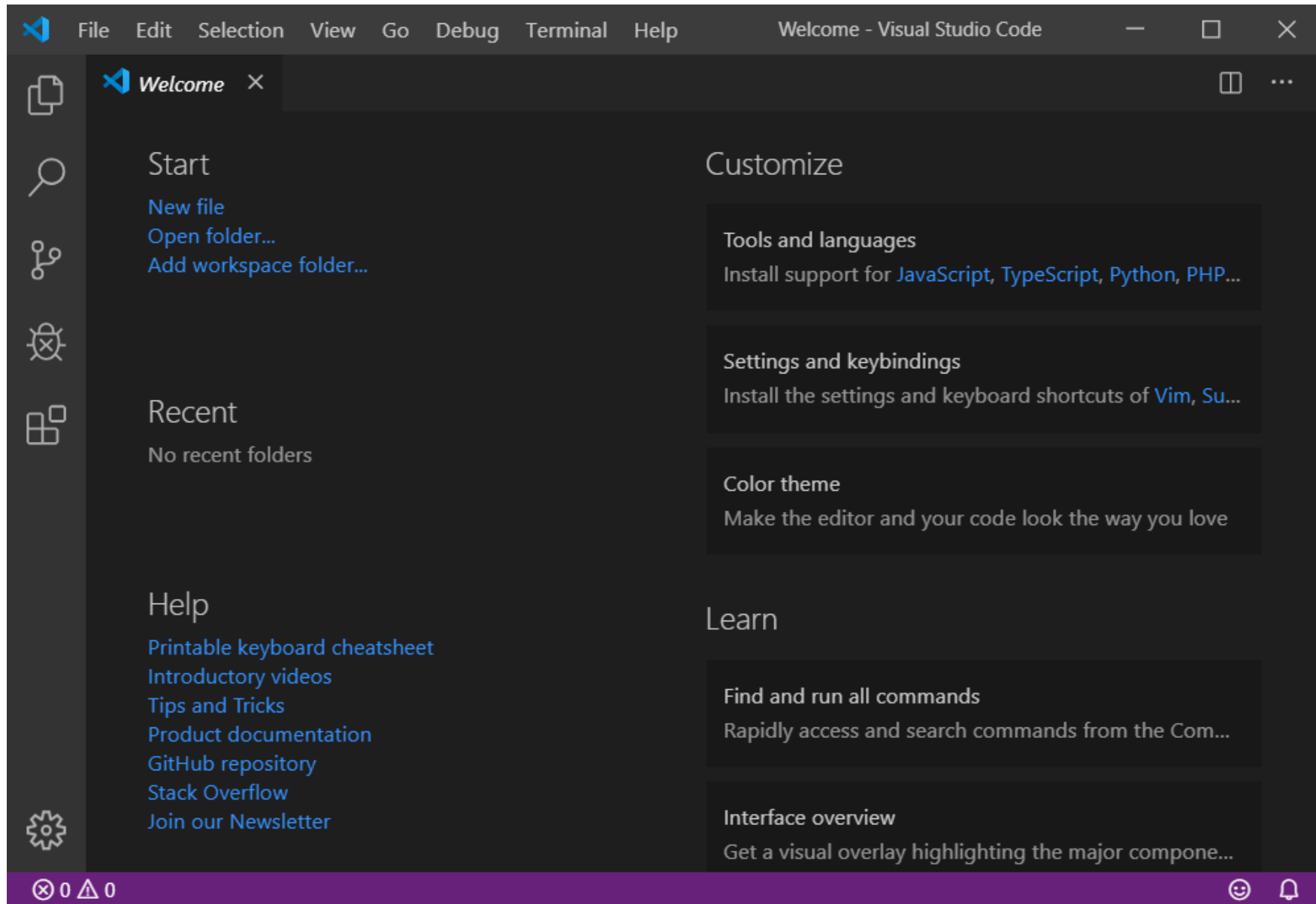
↓ Mac

macOS 10.10+

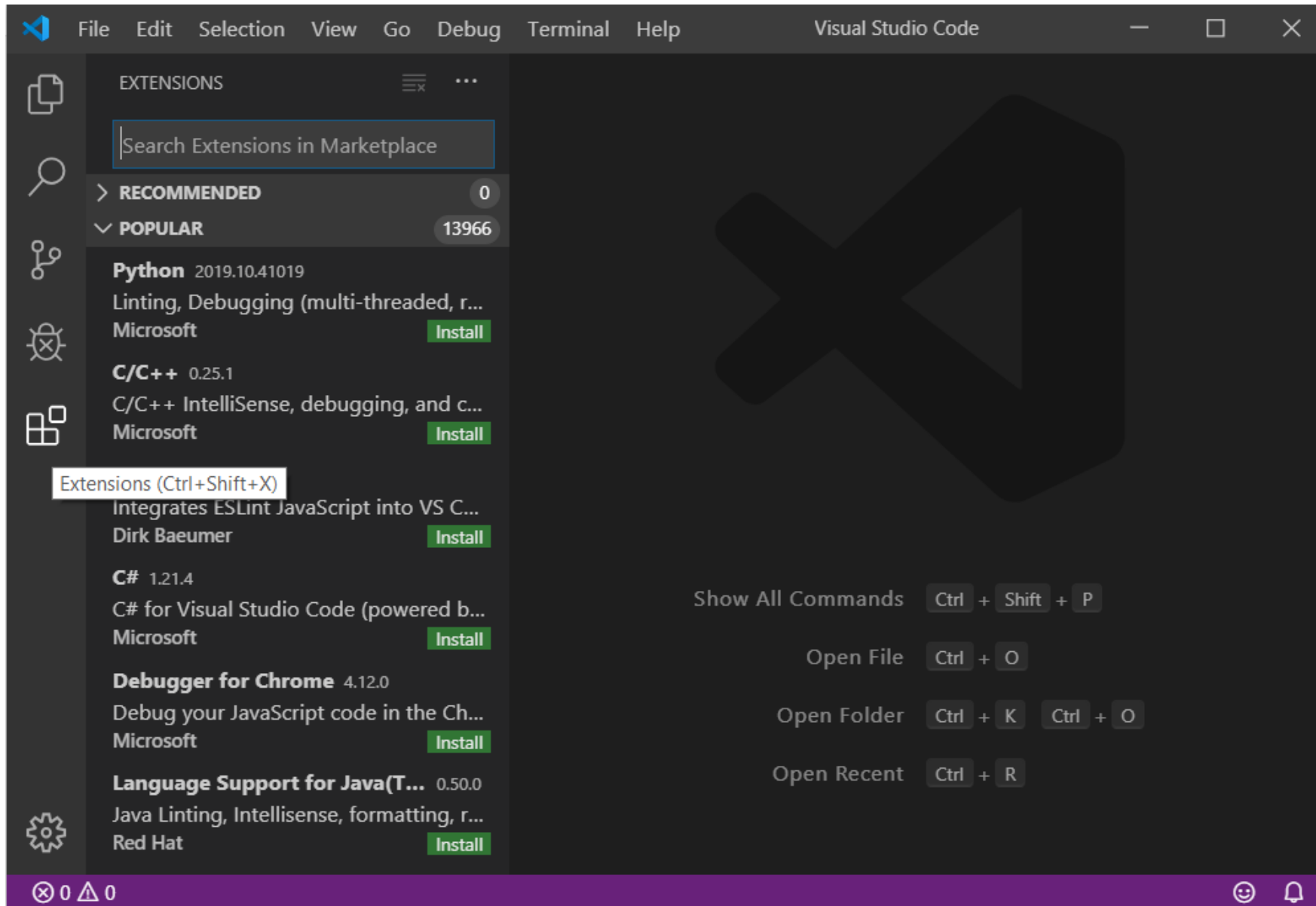
# VSCode (Cont.)



# VSCode (Cont.)



# VSCode Extensions





# VSCode Extensions (Cont.)

## ■ Open in Browser

- To open the current file in default browser.

## ■ Htmltagwrap

- Wraps selected code with HTML Tags

## ■ Vetur

- Vue.js code에 대한 문법 강조, 코드 자동완성, Debugging, Linting

## ■ Vue Peek

- Code 상에 작성되어 있는 Vue Component를 통해 해당 Component 파일을 열고 그 파일로 이동할 수 있는 기능

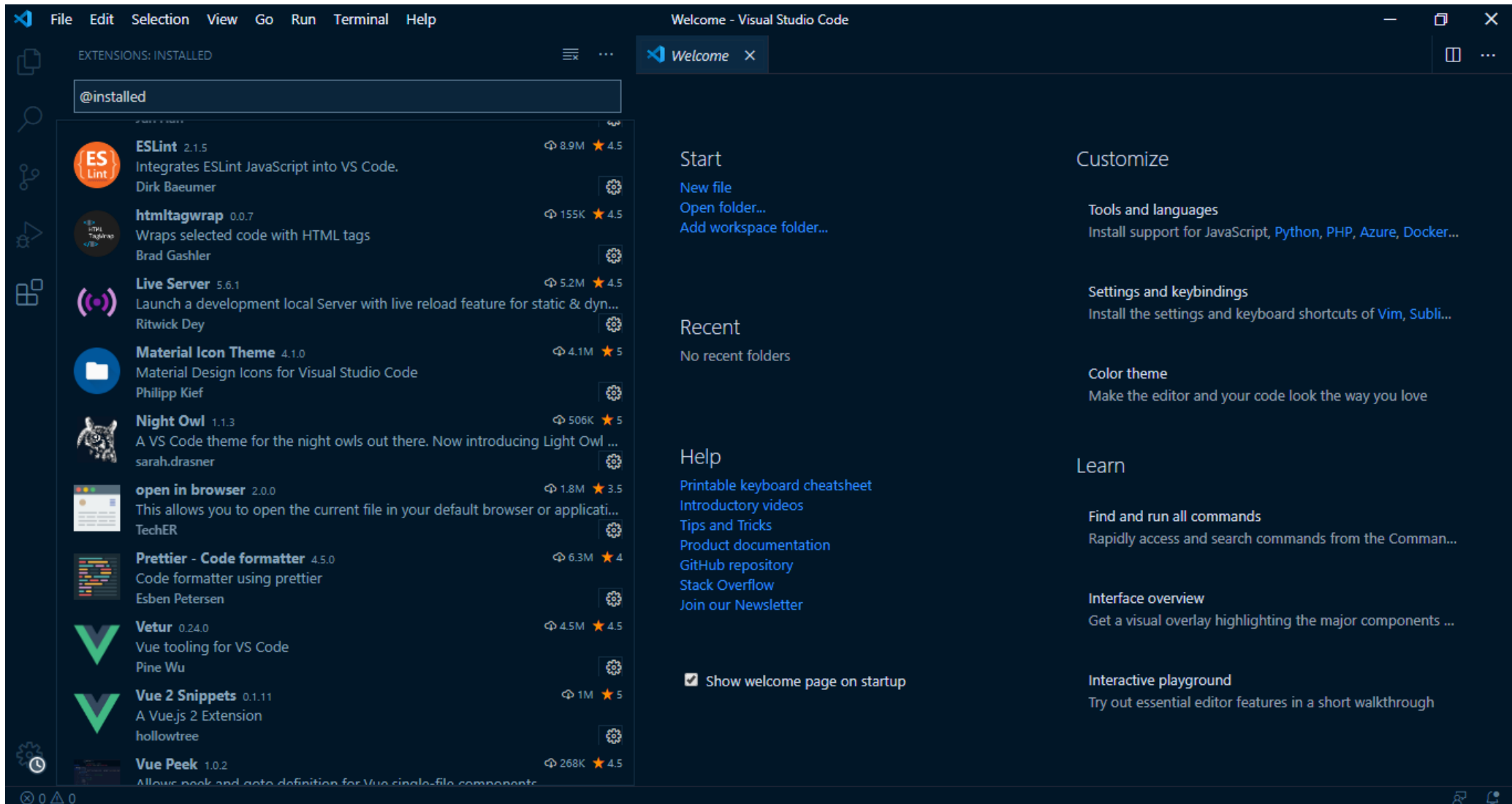
## ■ Vue 2 Snippets

- Vue Component 작성시 수기로 작성해야 할 부분을 VSCode의 Plugin에서 자동완성

# VSCode Extensions (Cont.)

- Night Owl
  - VSCode color theme.
- Material Icon Theme
  - VSCode icon theme
- Live Server
  - Launch a development local Server with live reload feature for static & dynamic pages.(Warning. Oracle 5500 port conflicting)
- ESLint
  - JavaScript code style, 문법 check 기능 제공
- Prettier
  - Code formatter
- Auto Close Tag

# VSCode Extensions (Cont.)

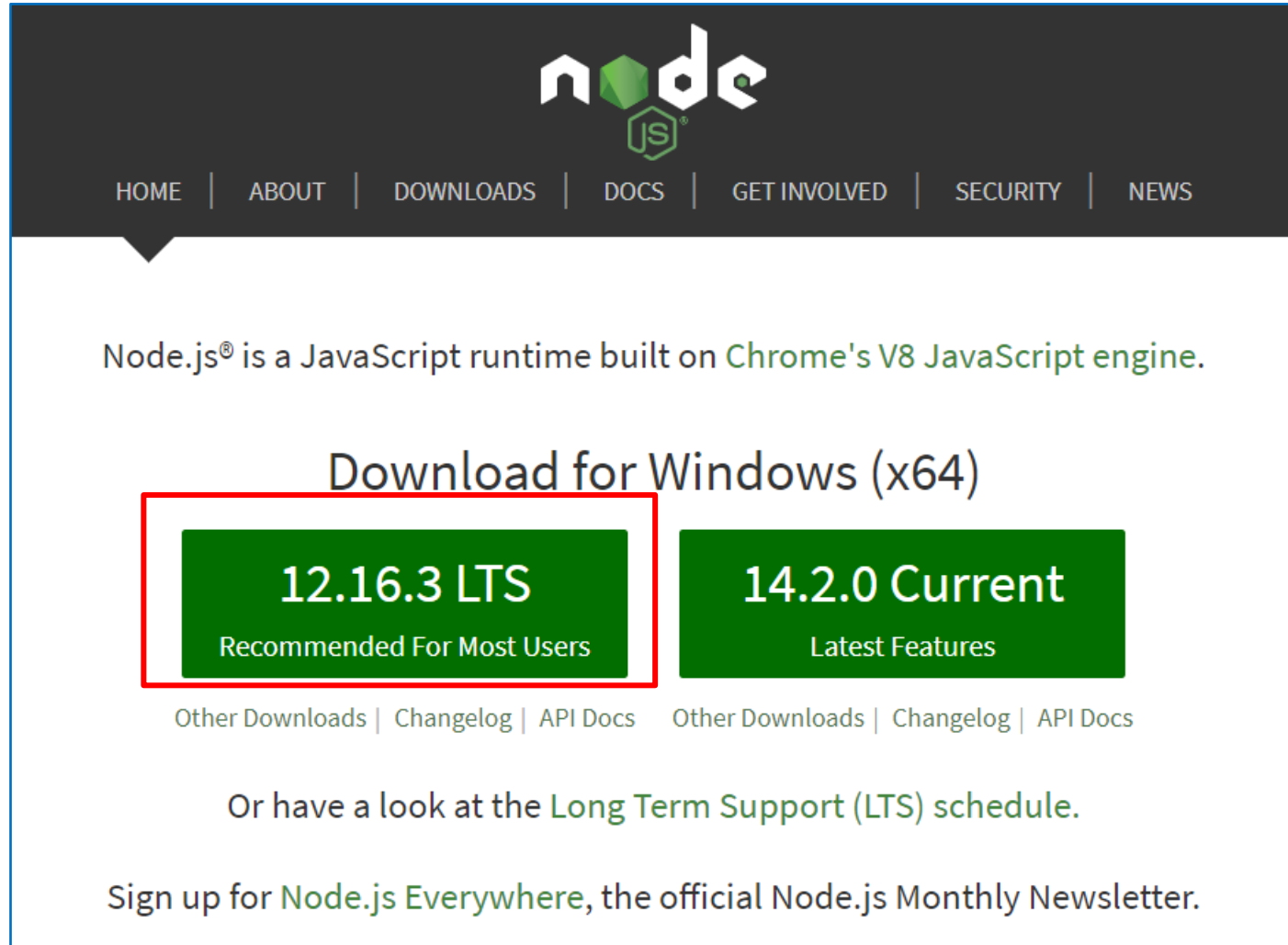


# VSCode Extensions (Cont.)

- [VSCode Theme for 2020](#)
- [8 Awesome Themes for Visual Studio Code](#)
- [6 Hottest VS Code Themes To Use in 2019](#)
- [VSCodethemes](#)

# Node.js

- <https://nodejs.org/en/>



The screenshot shows the Node.js website homepage. At the top is a dark navigation bar with the Node.js logo and links for HOME, ABOUT, DOWNLOADS, DOCS, GET INVOLVED, SECURITY, and NEWS. Below the navigation bar, the text states: "Node.js® is a JavaScript runtime built on Chrome's V8 JavaScript engine." The main section is titled "Download for Windows (x64)" and features two green buttons. The left button, labeled "12.16.3 LTS" and "Recommended For Most Users", is highlighted with a red rectangular border. The right button is labeled "14.2.0 Current" and "Latest Features". Below these buttons are links for "Other Downloads", "Changelog", and "API Docs" for both versions. At the bottom, there is a link to the "Long Term Support (LTS) schedule" and a sign-up link for "Node.js Everywhere", the official Node.js Monthly Newsletter.

node

HOME | ABOUT | DOWNLOADS | DOCS | GET INVOLVED | SECURITY | NEWS

Node.js® is a JavaScript runtime built on Chrome's V8 JavaScript engine.

Download for Windows (x64)

**12.16.3 LTS**  
Recommended For Most Users

**14.2.0 Current**  
Latest Features

Other Downloads | Changelog | API Docs    Other Downloads | Changelog | API Docs

Or have a look at the Long Term Support (LTS) schedule.

Sign up for Node.js Everywhere, the official Node.js Monthly Newsletter.

## Node.js (Cont.)

```
C:\Users\instructor>echo %PATH%  
C:\Windows\system32;C:\Windows;C:\Windows\System32\Wbem;C:\Windows\System32\Windows  
PowerShell\v1.0\;C:\Windows\System32\OpenSSH\;C:\Program Files\Microsoft VS Code\bin;  
C:\Program Files\nodejs\;C:\Users\instructor\AppData\Local\Microsoft\WindowsApps;  
C:\Users\instructor\AppData\Roaming\npm  
  
C:\Users\instructor>
```

```
C:\Users\instructor>  
C:\Users\instructor>node -v  
v12.16.3  
  
C:\Users\instructor>npm -v  
6.14.4
```

# Git

■ <https://git-scm.com/>



The screenshot shows the Git website homepage. At the top left is the Git logo (an orange diamond with a white branching diagram) followed by the text "git --everything-is-local". To the right is a search bar with the placeholder text "Search entire site...". Below the logo, there are two paragraphs of text describing Git. To the right of the text is a diagram showing a branching model with stacks of code blocks connected by colored lines (red, blue, yellow). At the bottom, there are four circular icons with corresponding text: a gear for "About", an open book for "Documentation", a download arrow for "Downloads", and two speech bubbles for "Community". On the right side of the bottom section is a monitor displaying the latest source release information.

**git** --everything-is-local

Search entire site...

Git is a **free and open source** distributed version control system designed to handle everything from small to very large projects with speed and efficiency.

Git is **easy to learn** and has a **tiny footprint with lightning fast performance**. It outclasses SCM tools like Subversion, CVS, Perforce, and ClearCase with features like **cheap local branching**, convenient **staging areas**, and **multiple workflows**.

**About**  
The advantages of Git compared to other source control systems.

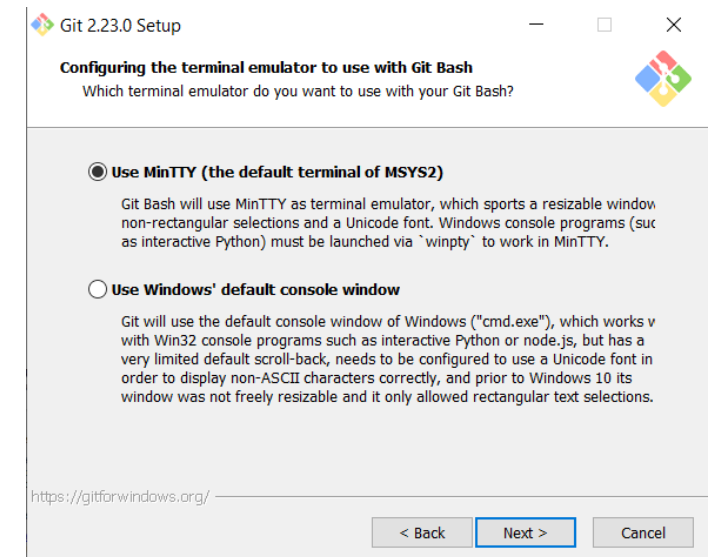
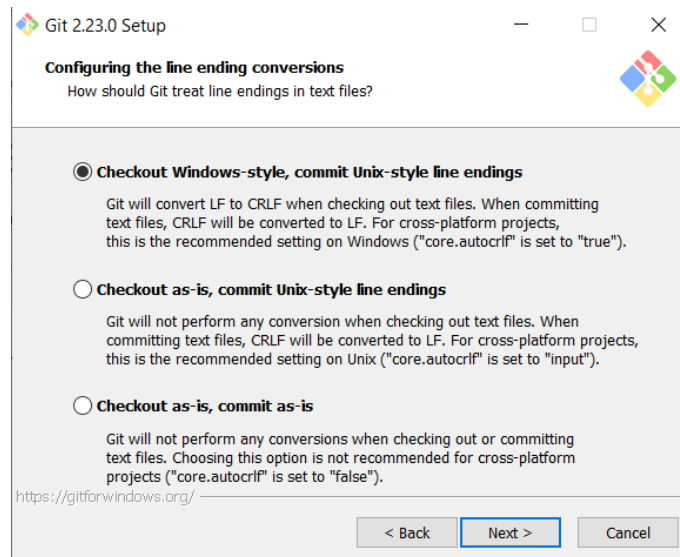
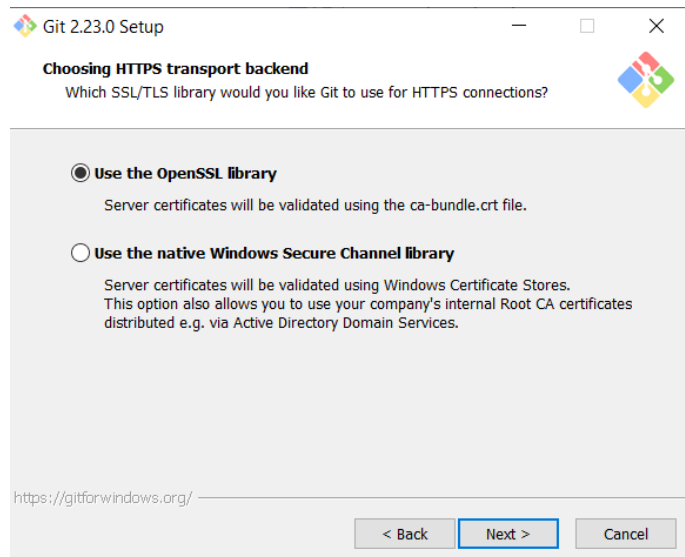
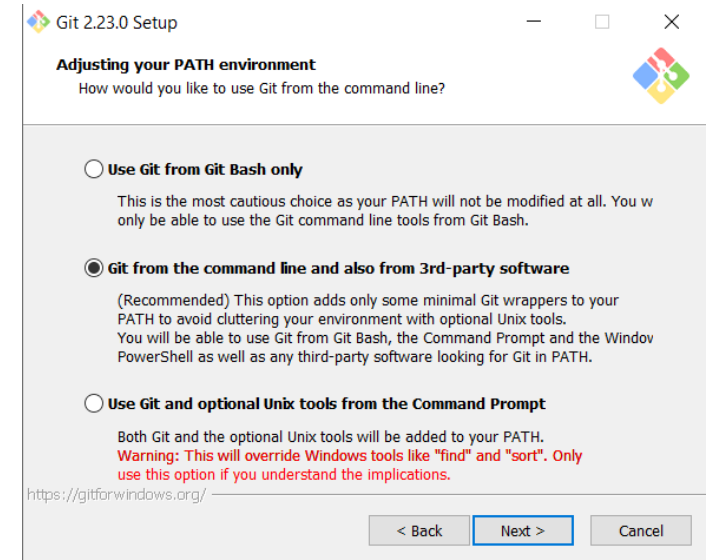
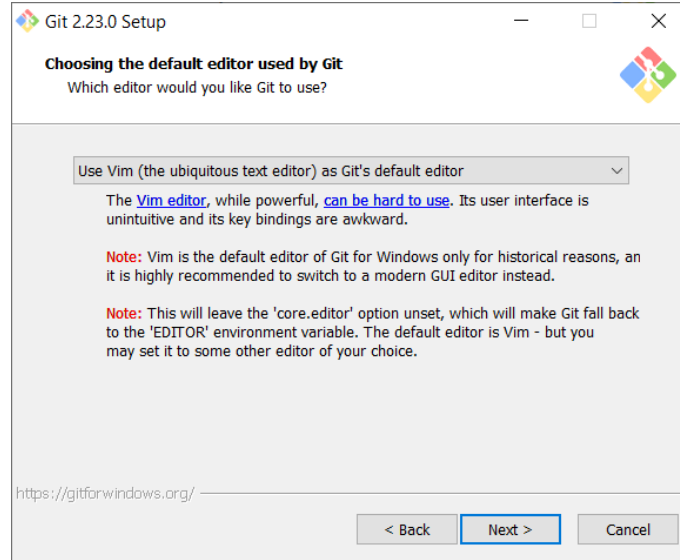
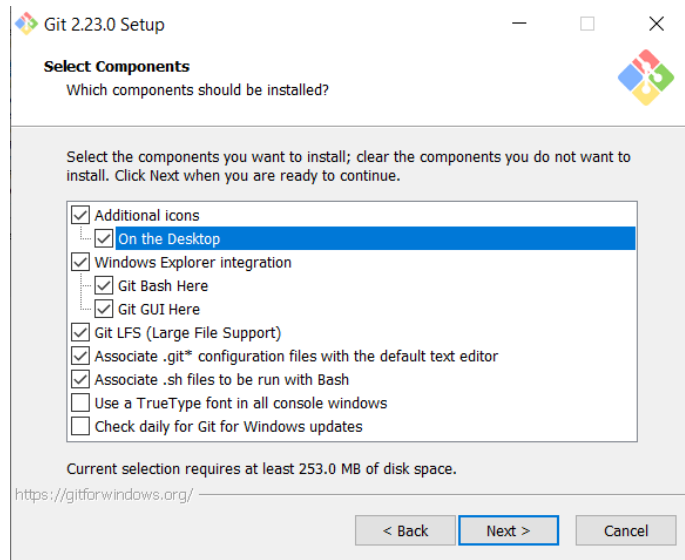
**Documentation**  
Command reference pages, Pro Git book content, videos and other material.

**Downloads**  
GUI clients and binary releases for all major platforms.

**Community**  
Get involved! Bug reporting, mailing list, chat, development and more.

Latest source Release  
**2.26.2**  
Release Notes (2020-04-19)  
Download 2.26.2 for Windows

# Git (Cont.)





# Git (Cont.)

Git 2.26.2 Setup



## Completing the Git Setup Wizard

Setup has finished installing Git on your computer. The application may be launched by selecting the installed shortcuts.

Click Finish to exit Setup.

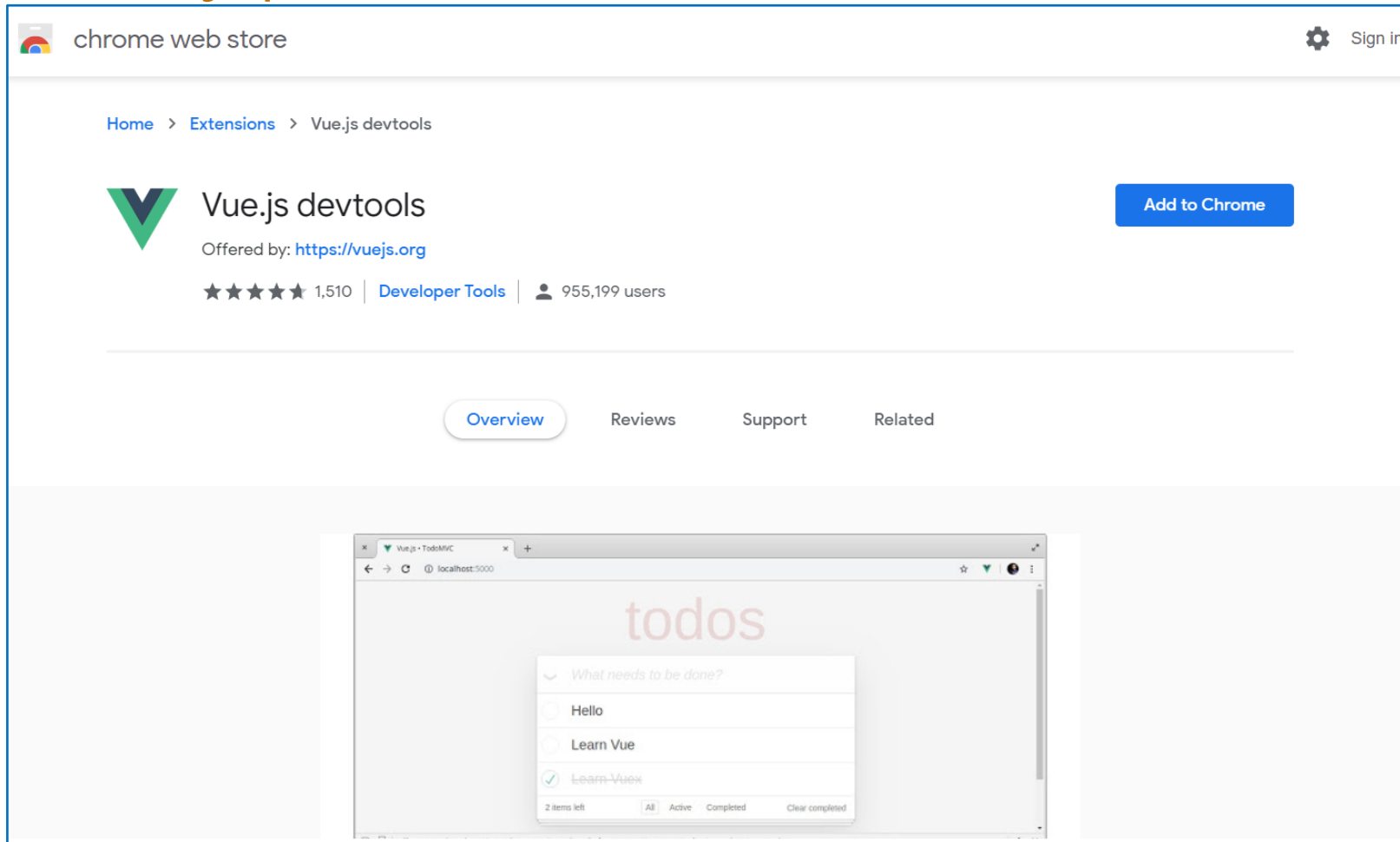
- ☐ Launch Git Bash
- ☒ View Release Notes

```
C:\Users\instructor>git --version  
git version 2.26.2.windows.1
```

Next >

# Vue.js devtools

- <https://chrome.google.com/webstore/detail/vuejs-devtools/nhdogjmejiglipccpnnnanhbledajbpd>



# Hello, Vue.js – First Way

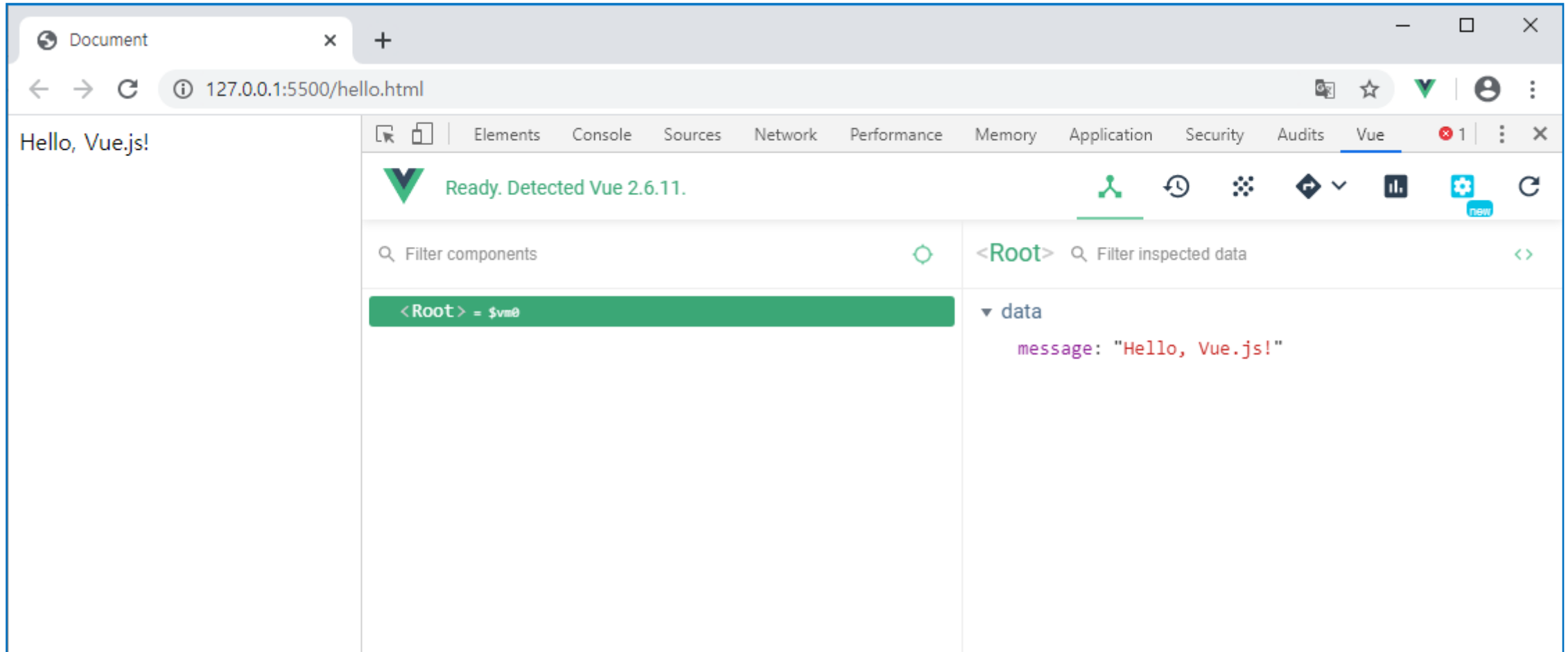
## ■ Stand Alone Type

- Direct **<script>** Include
- Simply download and include with a script tag.
- Vue will be registered as a global variable.
- Vuejs.org에서 Download
- Development Version
  - <https://vuejs.org/js/vue.js>
  - 342,147 Bytes
- Production Version
  - <https://vuejs.org/js/vue.min.js>
  - Vue.js devtools not showing in Chrome. Ref) <https://github.com/vuejs/vue-devtools>
  - 93,670 Bytes

# Hello, Vue.js – First Way (Cont.)

```
hello.html > html
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4    <meta charset="UTF-8">
5    <meta name="viewport" content="width=device-width, initial-scale=1.0">
6    <title>Document</title>
7  </head>
8  <body>
9    <div id="app">
10     {{ message }}
11   </div>
12   <script src="js/vue.js"></script>
13   <script>
14     new Vue({
15       el : '#app',
16       data : {
17         message : 'Hello, Vue.js!'
18       }
19     });
20   </script>
21 </body>
22 </html>
```

# Hello Vue.js – First Way (Cont.)



# Hello, Vue.js – Second Way

## ■ CDN Version

- Direct **<script>** Include
- Vue will be registered as a global variable.
- **<script src="https://cdn.jsdelivr.net/npm/vue/dist/vue.js"> </script>**
- **<script src="https://cdn.jsdelivr.net/npm/vue@2.6.12"> </script>**
- **<script type="module">**  
    **import Vue from 'https://cdn.jsdelivr.net/npm/vue@2.6.12/dist/vue.esm.browser.js'**  
    **</script>**
- You can browse the source of the NPM package at [cdn.jsdelivr.net/npm/vue](https://cdn.jsdelivr.net/npm/vue).

# Hello, Vue.js – Second Way (Cont.)

## ■ CDN Version

- Vue is also available on [unpkg](#) and [cdnjs](#).
- cdnjs takes some time to sync so the latest release may not be available yet.
- Make sure to :
  - Read about [the different builds of Vue](#)
  - Use the production version.
  - Replacing vue.js with vue.min.js.

# Hello, Vue.js – Second Way (Cont.)

```
hello.html
1  <!DOCTYPE html>
2  <html lang="en">
3  <head>
4    <meta charset="UTF-8">
5    <meta name="viewport" content="width=device-width, initial-scale=1.0">
6    <title>Document</title>
7  </head>
8  <body>
9    <div id="app">
10     {{ message }}
11   </div>
12   <!-- development version, includes helpful console warnings -->
13   <!-- <script src="https://cdn.jsdelivr.net/npm/vue@2.6.11/dist/vue.js"></script>-->
14
15   <!-- production version, optimized for size and speed -->
16   <script src="https://cdn.jsdelivr.net/npm/vue@2.6.11"></script>
17   </script>
18   new Vue({
19     el : '#app',
20     data : {
21       message : 'Hello, Vue.js!'
22     }
23   });
24   </script>
25 </body>
26 </html>
```



# Vue.js 개발 환경 선택하기

- Web Page에 Vue.js Library File을 포함하는 방법
- 배포용 Version
  - 개발자가 알아볼 수 없도록 난독화와 압축을 한 파일
  - 크기가 적고 파일을 편집기로 열었을 때 알아볼 수 없음.
  - Vue.js Project 개발을 완료하고 Web Application을 공개/배포 때 사용
  - vue.min.js
- 개발용 Version
  - Framework 내부에서 발생하는 경고 및 Debug Mode 모두 포함
  - 난독화하지 않은 File이므로 Vue.js Project 개발하면서 Vue.js 내부 동작을 살펴보기에 적합
  - vue.js
  - *Vue Devtools 사용가능*

## Vue.js 개발 환경 선택하기 (Cont.)

- Web page에 Vue.js Library file을 포함할 때는 CDN 사용 권장.
- 대표적인 CDN
  - <https://unpkg.com>
  - <https://www.jsdelivr.com>
  - <https://cdnjs.com>
- CDN에서 제공하는 Library File은 Version별로 구분해서 제공됨.
- NPM에 올라온 최신 Version을 사용할 때는 unpkg.com의 CDN 권장.

# Vue.js 개발 환경 선택하기 (Cont.)

## ■ unpkg.com

- 개발용 : <https://unpkg.com/vue@<version>/dist/vue.js>
- 배포용 : <https://unpkg.com/vue@<version>/dist/vue.min.js>
- 개발 Version에 한정해 <https://unpkg.com/vue>로 적용 가능

## ■ jsDelivr.com

- 개발용 : <https://cdn.jsdelivr.net/npm/vue@<version>/dist/vue.js>
- 배포용 : <https://cdn.jsdelivr.net/npm/vue@<version>/dist/vue.min.js>

## ■ cdnjs.com

- 개발용 : <https://cdnjs.cloudflare.com/ajax/libs/vue/<version>/vue.js>
- 배포용 : <https://cdnjs.cloudflare.com/ajax/libs/vue/<version>/vue.min.js>

## ■ Vue.js Project 개발시 개발 편의성을 위해 개발용 Version 권장

# Vue.js 개발 환경 선택하기 (Cont.)

## ■ NPM 설치

- Vue.js 개발시 ES6를 사용하거나 Vue.js 프로그램을 쉽게 배포하기 위한 Bundling할 때
- npm을 사용해서 Vue.js를 설치하면 *webpack*, *Browserify*와 같은 Module Bundler를 사용할 수 있다.
- 대규모 Web Application을 개발하기에 좋다.
- npm 명령어로 Vue.js를 설치하는 것은 Node.js Project에 적용할 때만 사용해야 한다.

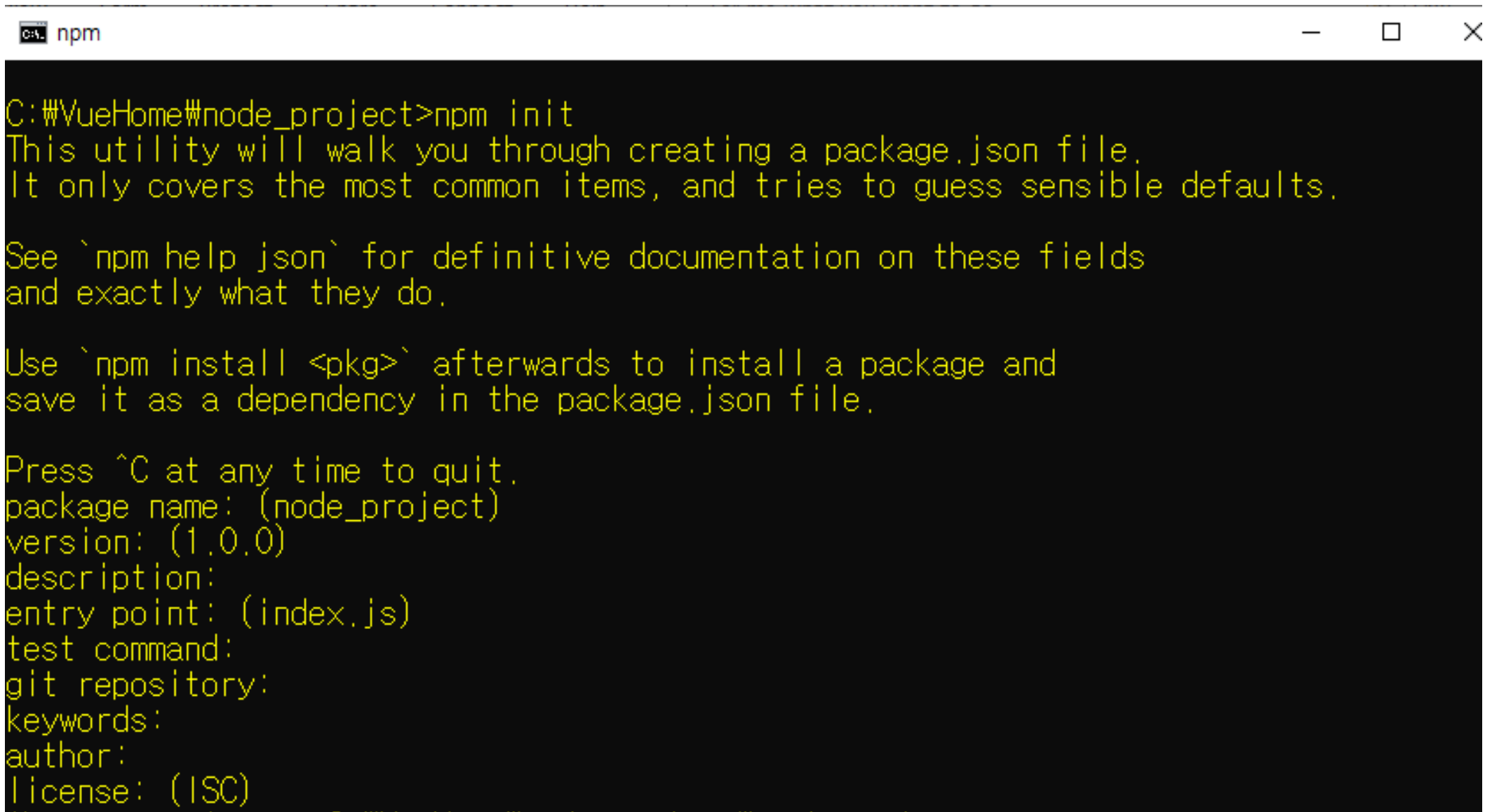
```
$ mkdir node_project
```

```
$ cd node_project
```

```
$ npm init
```

```
$ npm install vue
```

# Vue.js 개발 환경 선택하기 (Cont.)



```
cmd npm
C:\#VueHome#node_project>npm init
This utility will walk you through creating a package.json file.
It only covers the most common items, and tries to guess sensible defaults.

See `npm help json` for definitive documentation on these fields
and exactly what they do.

Use `npm install <pkg>` afterwards to install a package and
save it as a dependency in the package.json file.

Press ^C at any time to quit.
package name: (node_project)
version: (1.0.0)
description:
entry point: (index.js)
test command:
git repository:
keywords:
author:
license: (ISC)
```

## Vue.js 개발 환경 선택하기 (Cont.)

```
About to write to C:\#VueHome#node_project#package.json:
```

```
{
  "name": "node_project",
  "version": "1.0.0",
  "description": "",
  "main": "index.js",
  "scripts": {
    "test": "echo #\"Error: no test specified#\" && exit 1"
  },
  "author": "",
  "license": "ISC"
}
```

```
Is this OK? (yes) _
```

## Vue.js 개발 환경 선택하기 (Cont.)

```
C:\#VueHome#node_project>npm install vue
npm notice created a lockfile as package-lock.json. You should commit this file.
npm WARN node_project@1.0.0 No description
npm WARN node_project@1.0.0 No repository field.

+ vue@2.6.11
added 1 package from 1 contributor and audited 1 package in 1.283s
found 0 vulnerabilities
```

# Vue.js 개발 환경 선택하기 (Cont.)

## ■ Vue-CLI 설치

- Vue.js를 사용할 때 Vue 전용 CLI 도구인 Vue-CLI를 활용할 수 있다.
- Vue-CLI는 Node.js에 기반을 둔 Vue.js 프로그램의 빠른 시작과 구현을 위해 Build 설정이 포함된 Project를 생성한다.
- Vue-CLI 덕분에 Vue.js가 적용된 Node.js Project를 쉽게 만들 수 있다.
- Vue-CLI는 Project를 쉽게 Build할 수 있도록 몇 개의 Build Template을 제공한다.



# Vue-CLI Version 4

- Evan You가 공식적으로 관리하는 Command line Interface 기반의 Scaffolding 도구.
- Vue.js App을 개발시 Project의 Directory 구조를 어떻게 잡을 것인지 Test, Build, Dependency 등을 어떻게 설정할 것인지에 대한 고민을 하지 않도록 Project의 기본 Template을 설정해 주는 Tool.
- Refer to <https://cli.vuejs.org/guide/cli-service.html>
- Prerequisites:
  - Node.js (>=6.x, 8.x preferred), npm version 3+ and Git.
- Installation

**\$ npm install -g @vue/cli**

**\$ yarn global add @vue/cli**

**Scaffolding** : 원래 '공사장 등에서 사용하는 임시 가설물'이라는 뜻. Application을 개발할 때 처음부터 개발하는 것이 아니라 기본적인 Interface와 Tool을 제공해 준다. 개발자는 이 Tool을 활용해서 전체 App을 개발할 수 있다. - 원형섭, <Vue.js 쿼 스타트>, 2017.9, 루비페이퍼, p11.

## Vue-CLI Version 4 (Cont.)

### ■ Warning regarding Previous Versions

- The package name changed from **vue-cli** to **@vue/cli**.
- If have the previous **vue-cli** (1.x or 2.x) package installed globally, need to uninstall it first with :

**\$ npm uninstall vue-cli -g**

or

**\$ yarn global remove vue-cli**

# Vue-CLI Version 4 (Cont.)

```
C:\VueHome>npm install -g @vue/cli
npm WARN deprecated request@2.88.2: request has been deprecated, see https://github.com/request/request/issues/3142
npm WARN deprecated har-validator@5.1.5: this library is no longer supported
npm WARN deprecated urix@0.1.0: Please see https://github.com/lydell/urix#deprecated
npm WARN deprecated resolve-url@0.2.1: https://github.com/lydell/resolve-url#deprecated
npm WARN deprecated chokidar@2.1.8: Chokidar 2 will break on node v14+. Upgrade to chokidar 3 with 15x less dependencies

C:\Users\devex\AppData\Roaming\npm\vue -> C:\Users\devex\AppData\Roaming\npm\node_modules\@vue\cli\bin\vue.js

> @apollo/protobufjs@1.0.5 postinstall C:\Users\devex\AppData\Roaming\npm\node_modules\@vue\cli\node_modules\@apollo\pro
tobufjs
> node scripts/postinstall

npm WARN optional SKIPPING OPTIONAL DEPENDENCY: fsevents@^1.2.7 (node_modules\@vue\cli\node_modules\chokidar\node_modules
s\fsevents):
npm WARN notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@1.2.13: wanted {"os":"darwin","arch":"an
y"} (current: {"os":"win32","arch":"x64"})
npm WARN @vue/compiler-sfc@3.0.0-rc.10 requires a peer of vue@3.0.0-rc.10 but none is installed. You must install peer d
ependencies yourself.

+ @vue/cli@4.5.4
added 211 packages from 134 contributors, removed 67 packages, updated 240 packages and moved 8 packages in 81.888s
```

## Vue-CLI Version 4 (Cont.)

- After installation, will have access to the **vue** binary in command line.
- 설치 후 설치 유무 및 Version 정보 확인  
**\$ vue -V** 혹은 **vue --version**

```
C:\#VueHome>vue -V
@vue/cli 4.5.4
```

# Vue-CLI Version 4 (Cont.)

## ■ Project 생성

**\$ vue create hello-vue-cli**

```
C:\#VueHome>vue create hello-vue-cli
```

```
Vue CLI v4.5.4
```

```
? Please pick a preset: (Use arrow keys)
```

```
> Default ([Vue 2] babel, eslint)
```

```
Default (Vue 3 Preview) ([Vue 3] babel, eslint)
```

```
Manually select features
```

# Vue-CLI Version 4 (Cont.)

```
> ejs@2.7.4 postinstall C:\VueHome\hello-vue-cli\node_modules\ejs  
> node ./postinstall.js
```

```
added 1242 packages from 918 contributors in 150.943s
```

```
50 packages are looking for funding  
  run `npm fund` for details
```

```
❖ Invoking generators...  
❖ Installing additional dependencies...
```

```
added 53 packages from 36 contributors in 24.061s
```

```
54 packages are looking for funding  
  run `npm fund` for details
```

```
⚙ Running completion hooks...
```

```
❖ Generating README.md...
```

```
❖ Successfully created project hello-vue-cli.  
❖ Get started with the following commands:
```

```
$ cd hello-vue-cli  
$ npm run serve
```

```
C:\VueHome>
```

# Vue-CLI Version 4 (Cont.)

```
C:\VueHome>cd hello-vue-cli
C:\VueHome\hello-vue-cli>npm run serve
> hello-vue-cli@0.1.0 serve C:\VueHome\hello-vue-cli
> vue-cli-service serve
```

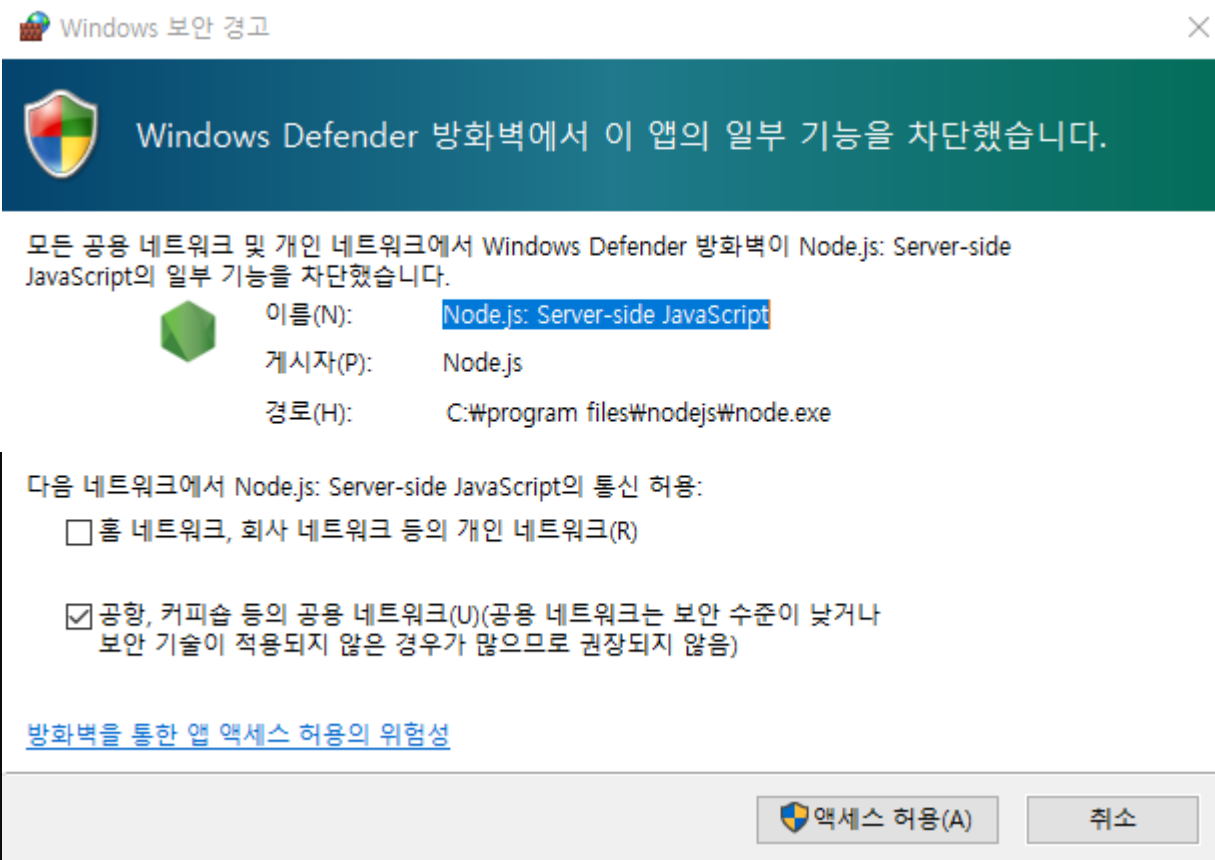
```
INFO Starting development server...
98% after emitting CopyPlugin
```

```
DONE Compiled successfully in 3373ms
```

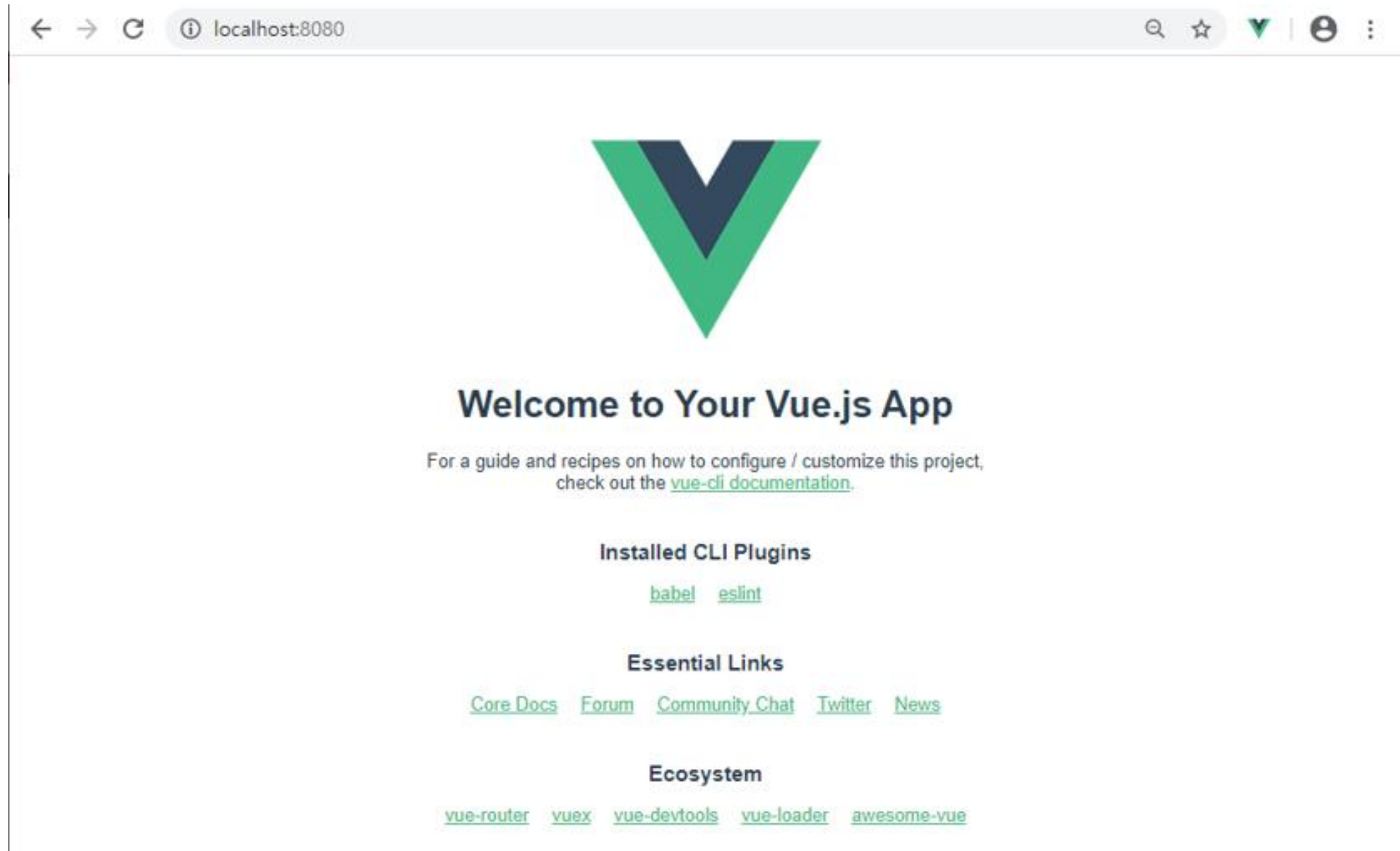
App running at:

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

Note that the development build is not optimized.  
To create a production build, run `npm run build`.

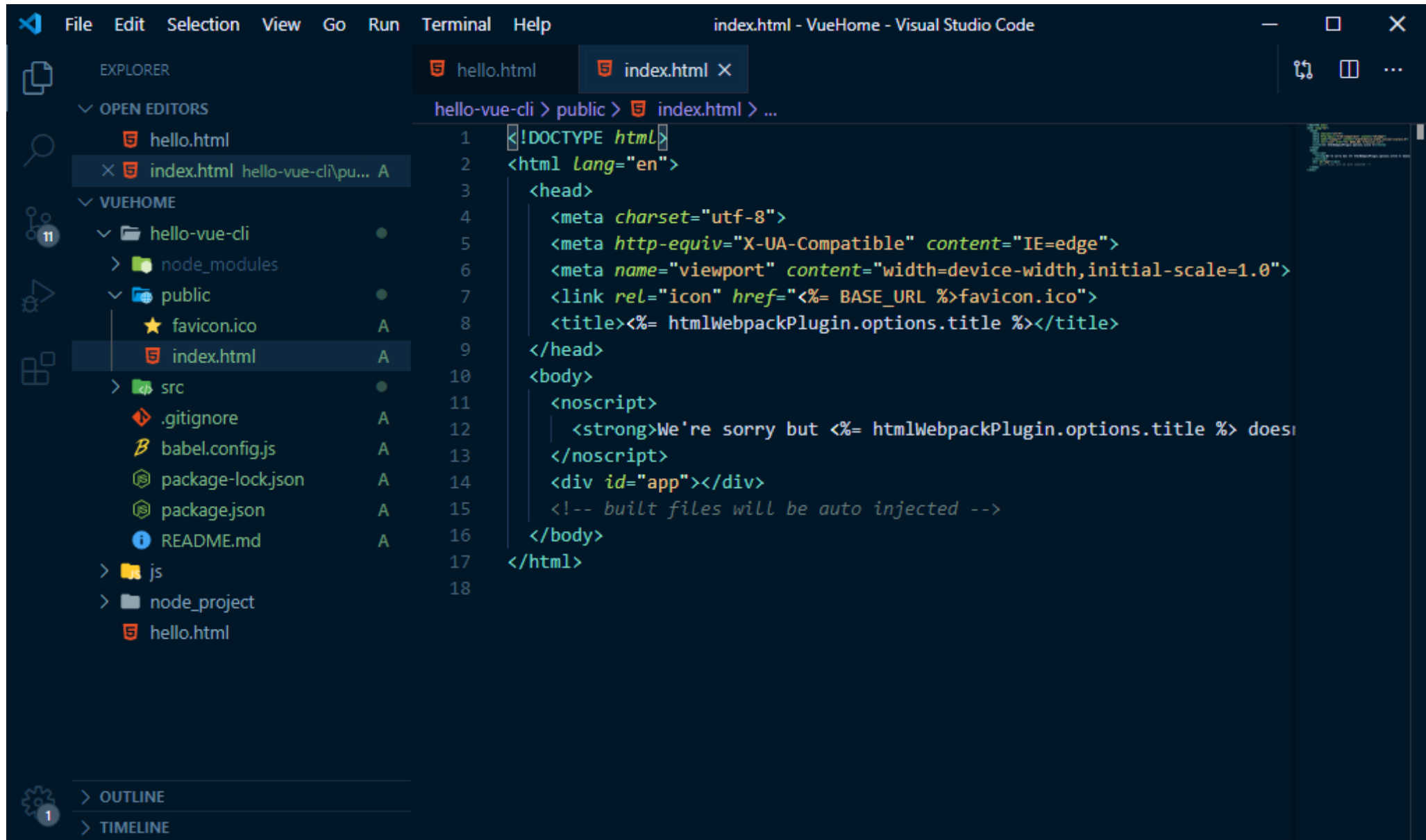


# Vue-CLI Version 4 (Cont.)





# Vue-CLI Version 4 (Cont.)



# Vue-CLI Version 2

## ■ Remove Vue-CLI 4.x

일괄 작업을 끝내시겠습니까 (Y/N)? y

```
C:\VueHome\hello-vue-cli>cd ..
```

```
C:\VueHome>npm remove -g @vue/cli
```

```
npm WARN optional SKIPPING OPTIONAL DEPENDENCY: fsevents@^1.2.7 (node_modules\@vue\cli\node_modules\chokidar\node_modules\fs\fs_events):
```

```
npm WARN notsup SKIPPING OPTIONAL DEPENDENCY: Unsupported platform for fsevents@1.2.13: wanted {"os":"darwin","arch":"any"} (current: {"os":"win32","arch":"x64"})
```

```
removed 1316 packages in 18.164s
```

```
C:\VueHome>vue -V
```

'vue'은(는) 내부 또는 외부 명령, 실행할 수 있는 프로그램, 또는  
배치 파일이 아닙니다.

```
C:\VueHome>
```

# Vue-CLI Version 2

## ■ Installation Vue-CLI 2.x

```
C:\#VueHome>npm install vue-cli -g
npm WARN deprecated vue-cli@2.9.6: This package has been deprecated in favour of @vue/cli
npm WARN deprecated request@2.88.2: request has been deprecated, see https://github.com/request/request/issues/3142
npm WARN deprecated coffee-script@1.12.7: CoffeeScript on NPM has moved to "coffeescript" (no hyphen)
C:\#Users\#instructor\AppData\Roaming\npm#vue-init -> C:\#Users\#instructor\AppData\Roaming\npm\node_modules\vue-cli\bin\vue-init
C:\#Users\#instructor\AppData\Roaming\npm#vue -> C:\#Users\#instructor\AppData\Roaming\npm\node_modules\vue-cli\bin\vue
C:\#Users\#instructor\AppData\Roaming\npm#vue-list -> C:\#Users\#instructor\AppData\Roaming\npm\node_modules\vue-cli\bin\vue-list
+ vue-cli@2.9.6
added 238 packages from 205 contributors in 62.098s

C:\#VueHome>_
```

## Vue-CLI Version 2 (Cont.)

- 설치 후 설치 유무 및 Version 정보 확인

```
C:\#VueHome>vue --version  
2.9.6
```

- Vue-CLI 2.x로 만들 수 있는 Project Template List

**\$ vue list**

## Vue-CLI Version 2 (Cont.)

### ■ Vue-CLI 2.x로 만들 수 있는 Project Template List

**\$ vue list**

```
C:\#VueHome>vue list
```

Available official templates:

- ★ browserify - A full-featured Browserify + vueify setup with hot-reload, linting & unit testing.
- ★ browserify-simple - A simple Browserify + vueify setup for quick prototyping.
- ★ pwa - PWA template for vue-cli based on the webpack template
- ★ simple - The simplest possible Vue setup in a single HTML file
- ★ webpack - A full-featured Webpack + vue-loader setup with hot reload, linting, testing & css extraction.
- ★ webpack-simple - A simple Webpack + vue-loader setup for quick prototyping.

```
C:\#VueHome>
```

# Vue-CLI Version 2 (Cont.)

## ■ simple

- 가장 간단단 Project Template 유형.
- HTML file 하나만 생성.
- 내부에는 Vue.js를 CDN(Content Delivery Network)방식으로 참조하고 있음.
- 가장 단순한 방법으로 기본적인 기능을 Test 해 볼 때 사용 가능.

## ■ broserify

- browserify + vueify 조합으로 hot-reload, linting, 단위 Test를 지원하는 Project Template을 구성.
- 대규모 Application을 개발할 때 주로 사용

## ■ broserify-simple

- 빠른 Prototyping을 위한 browserify + vueify가 설정된 간단한 Template 제공

# Vue-CLI Version 2 (Cont.)

## ■ webpack

- Webpack + vue-loader 조합의 기능에 hot-reload, linting, 단위 Test 기능을 제공하는 Project Template.
- 대규모 Application 개발에 적절.

## ■ webpack-simple

- 빠른 Prototyping을 위한 Webpack + vue-loader가 설정된 간단한 Template 제공

## ■ pwa

- PWA(Progressive Web App) 개발을 위한 Template 제공.
- Webpack과 유사하지만 몇 가지 Library가 추가되어 있음.

## Vue-CLI (Cont.)

- Vue-CLI 2.x로 Project Template을 만드는 방법은 아래와 같다.

**\$ vue init [Template이름] [Project 이름]**

- hellovuejs Project 생성하기

**\$ vue init simple hellovuejs**

```
C:\#VueHome>vue init simple hellovuejs
```

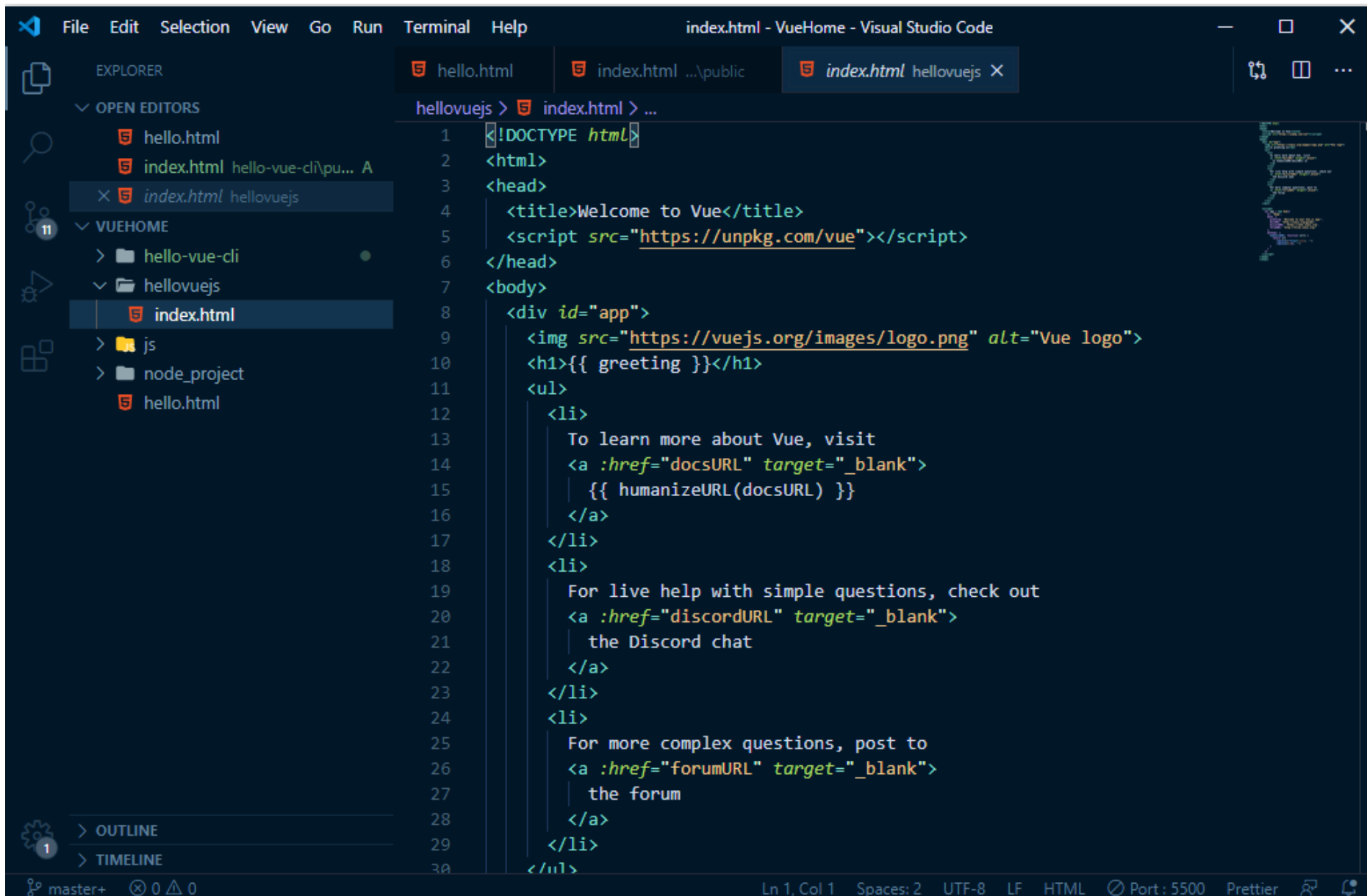
```
? name hellovuejs
```

```
vue-cli · Generated "hellovuejs".
```

```
C:\#VueHome>
```

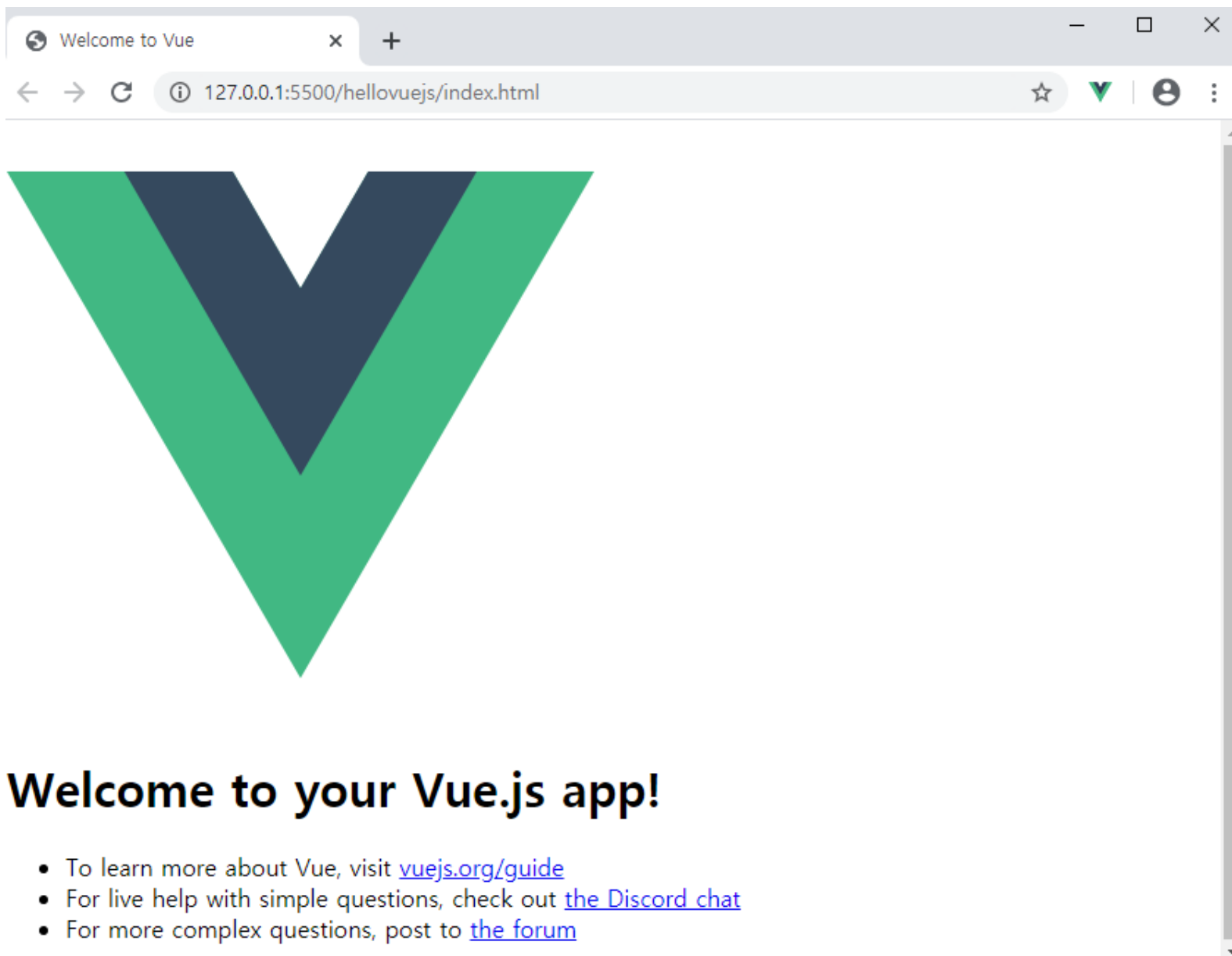


# Vue-CLI (Cont.)

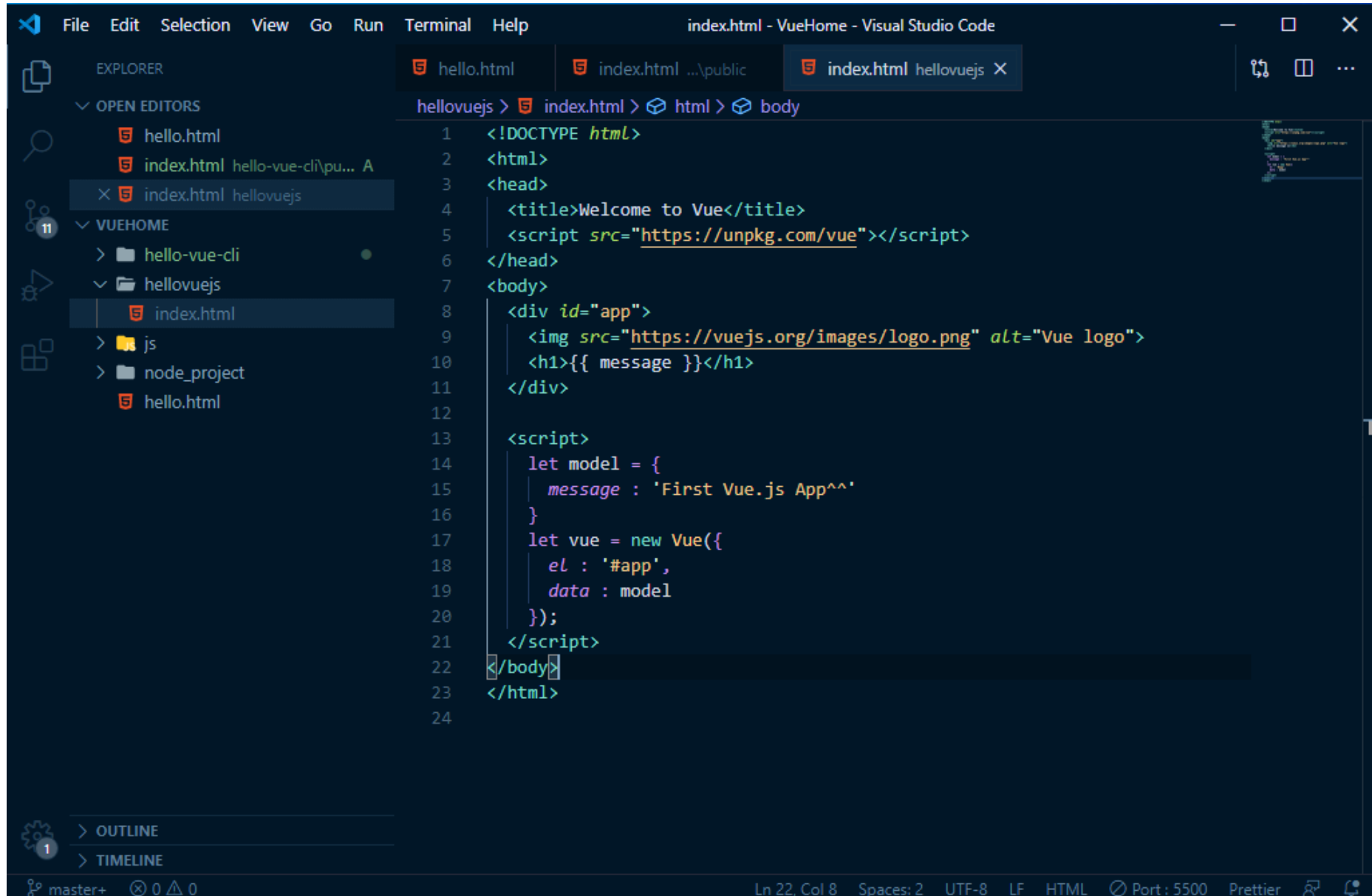


```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <title>Welcome to Vue</title>
5   <script src="https://unpkg.com/vue"></script>
6 </head>
7 <body>
8   <div id="app">
9     
10    <h1>{{ greeting }}</h1>
11    <ul>
12      <li>
13        To learn more about Vue, visit
14        <a :href="docsURL" target="_blank">
15          | {{ humanizeURL(docsURL) }}
16        </a>
17      </li>
18      <li>
19        For live help with simple questions, check out
20        <a :href="discordURL" target="_blank">
21          | the Discord chat
22        </a>
23      </li>
24      <li>
25        For more complex questions, post to
26        <a :href="forumURL" target="_blank">
27          | the forum
28        </a>
29      </li>
30    </ul>
```

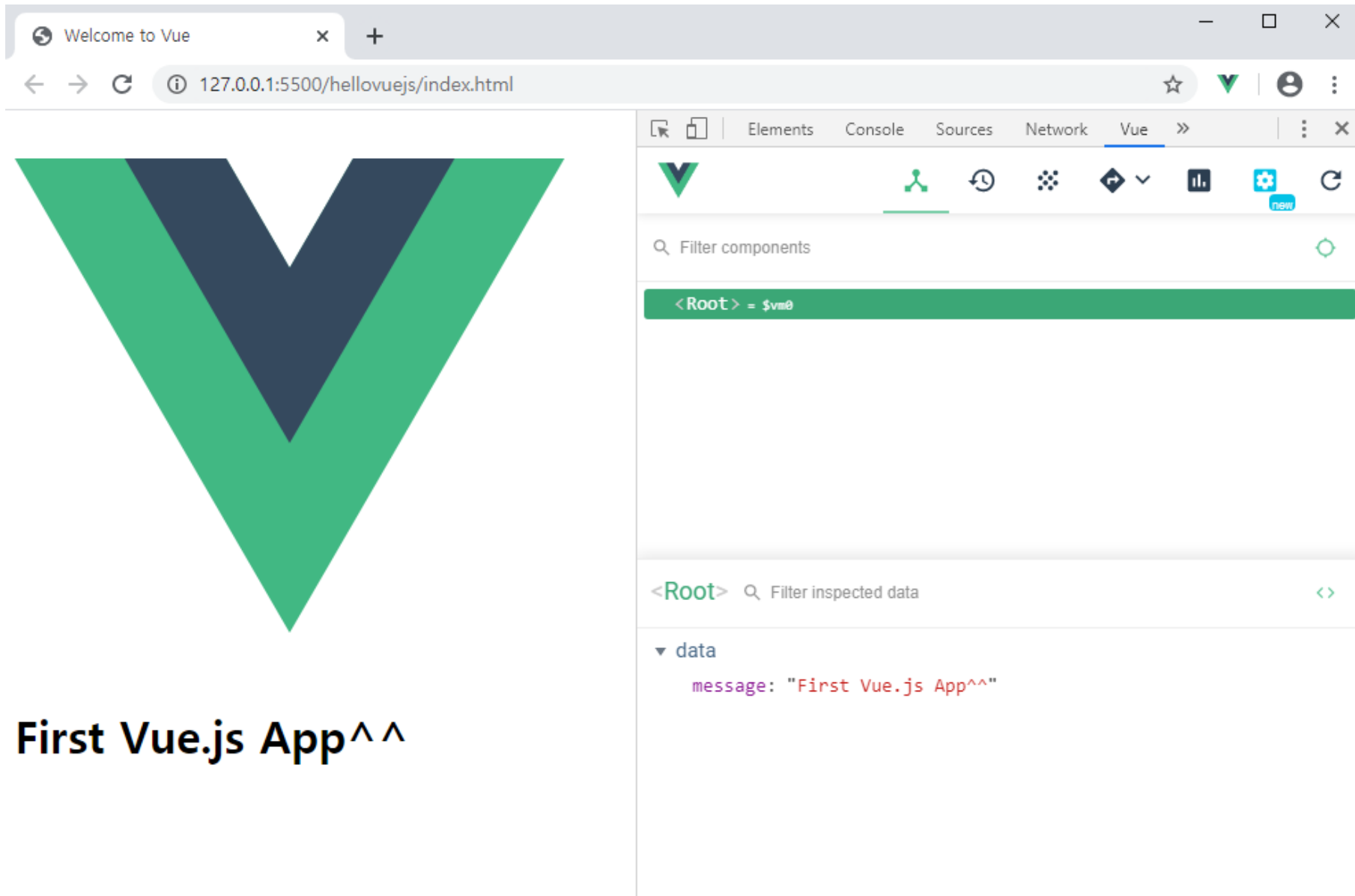
# Vue-CLI (Cont.)



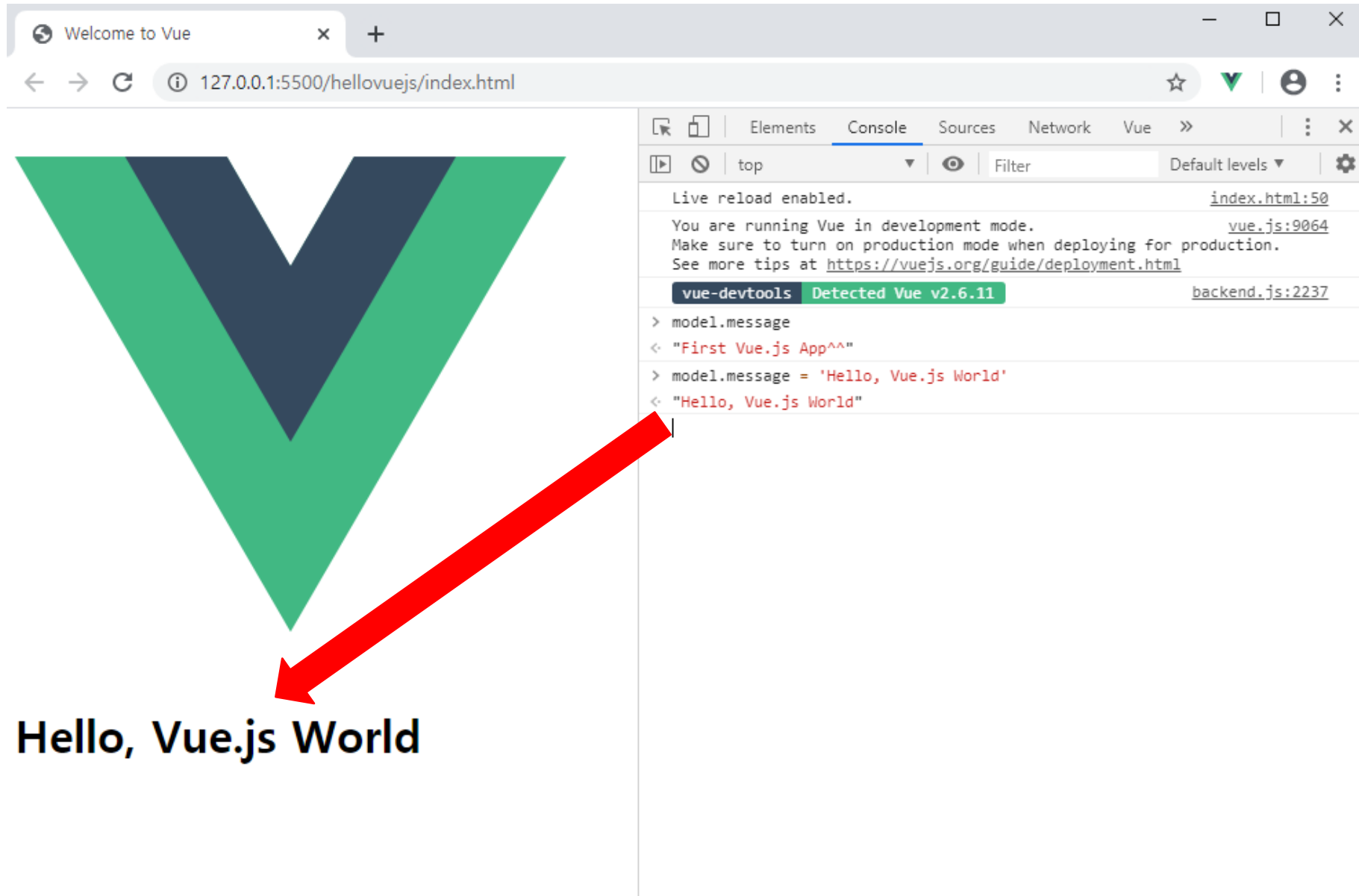
# Vue-CLI (Cont.)



# Vue-CLI (Cont.)



# Vue-CLI (Cont.)



# Vue-CLI (Cont.)

**\$ vue init webpack hello-world**

```
C:\#VueHome>vue init webpack hello-world

? Project name hello-world
? Project description A Vue.js project
? Author instructor
? Vue build standalone
? Install vue-router? Yes
? Use ESLint to lint your code? Yes
? Pick an ESLint preset Standard
? Set up unit tests Yes
? Pick a test runner jest
? Setup e2e tests with Nightwatch? Yes
? Should we run `npm install` for you after the project has been created? (recommended) npm

vue-cli · Generated "hello-world".

# Installing project dependencies ...
# =====
```

## Vue-CLI (Cont.)

**\$ vue init webpack hello-world**

```
Running eslint --fix to comply with chosen preset rules...
```

```
# =====
```

```
> hello-world@1.0.0 lint C:\#VueHome\#hello-world
```

```
> eslint --ext .js,.vue src test/unit test/e2e/specs "--fix"
```

```
# Project initialization finished!
```

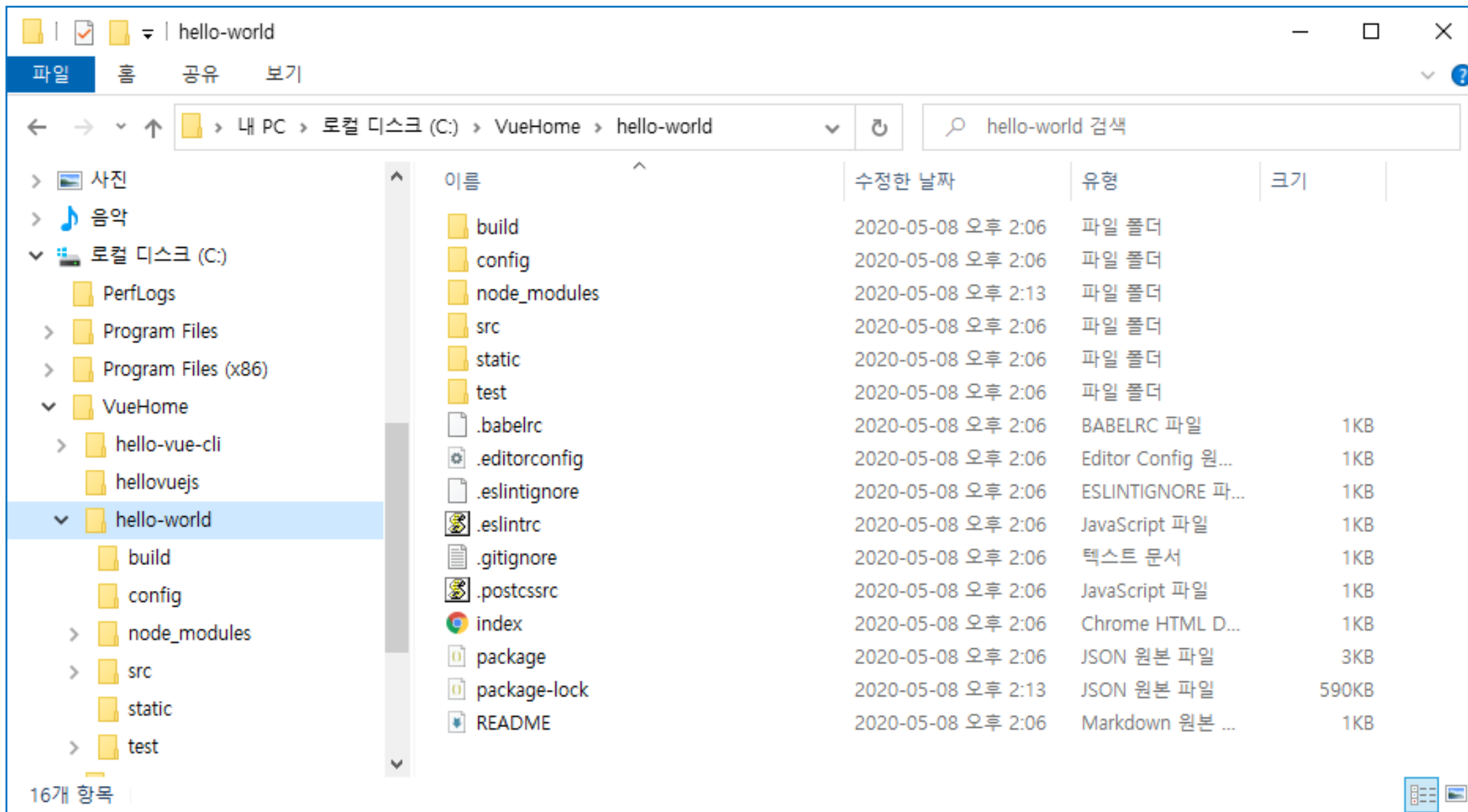
```
# =====
```

```
To get started:
```

```
  cd hello-world  
  npm run dev
```

```
Documentation can be found at https://vuejs-templates.github.io/webpack
```

# Vue-CLI (Cont.)





## Vue-CLI (Cont.)

```
C:\#VueHome>cd hello-world
```

```
C:\#VueHome#hello-world>npm run dev
```

```
> hello-world@1.0.0 dev C:\#VueHome#hello-world
```

```
> webpack-dev-server --inline --progress --config build/webpack.dev.conf.js
```

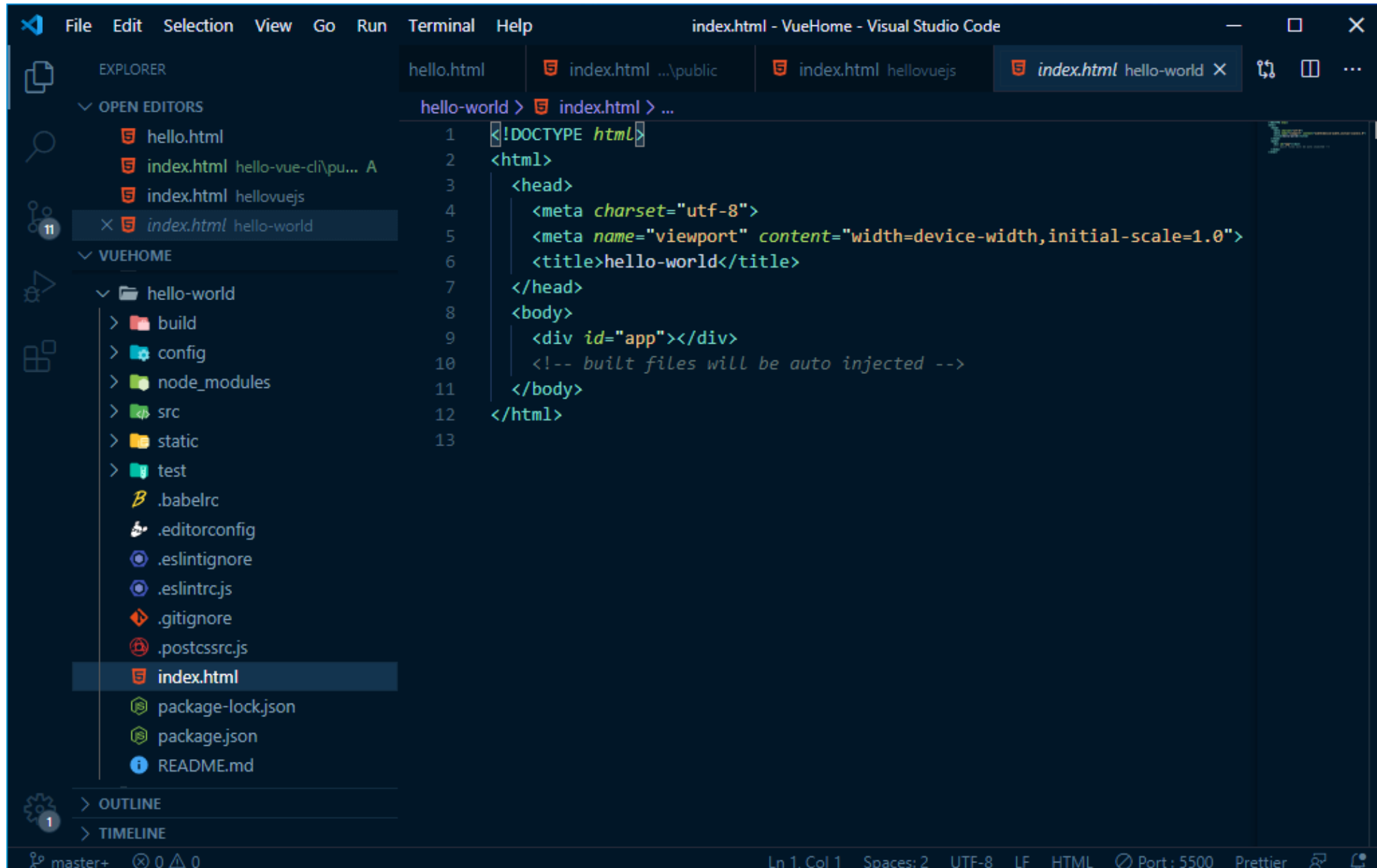
```
13% building modules 28/31 modules 3 active ...=0!C:\#VueHome#hello-world#src#App.vue{ parser: "babel" } is deprecated; we now treat it as { parser: "babel" }.  
95% emitting
```

```
DONE Compiled successfully in 5956ms
```

14:15:38

```
| Your application is running here: http://localhost:8080
```

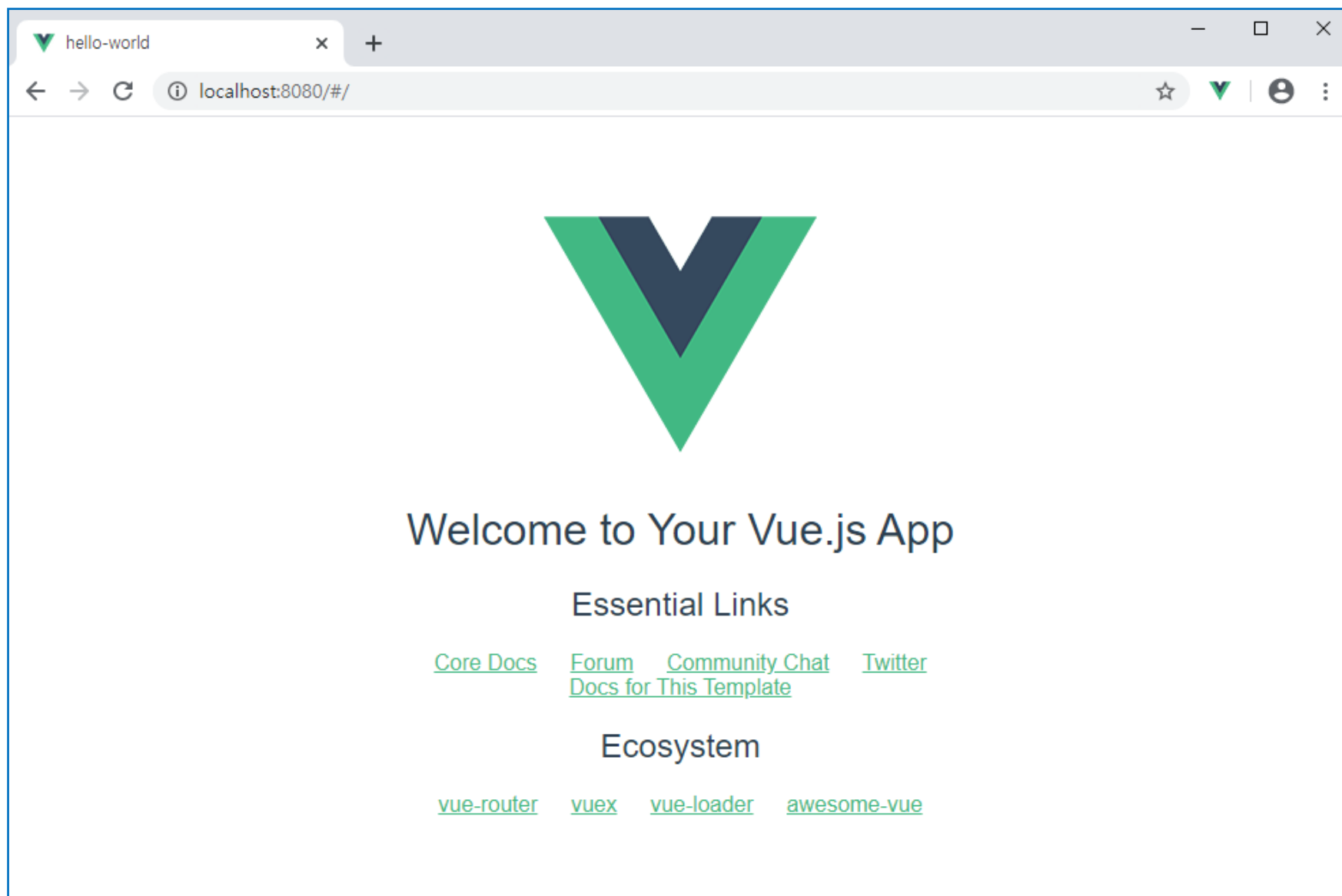
# Vue-CLI (Cont.)



# Vue-CLI (Cont.)



# Vue-CLI (Cont.)



# Online Editor – JSFiddle(https://jsfiddle.net)

The screenshot shows the JSFiddle website interface. At the top, there's a blue header bar with a 'Close' button, a 'Start with a boilerplate:' section featuring buttons for jQuery, Vue, React, React + JSX, Preact, TypeScript, CoffeeScript, SCSS, CSS Grid (responsive), Bootstrap, and PostCSS (marked as 'New'), a 'Links:' section with Roadmap, Bug tracker, About, Docs, and Service status, and a 'JSFiddle is for:' section listing various use cases. Below the header is a dark grey toolbar with 'Run', 'Save', 'Tidy', 'Collaborate', 'Update', and 'New and updated JS/CSS linters' buttons. The main workspace is divided into three panels: 'Fiddle meta' on the left with fields for title and description, 'HTML' in the center with a code editor, and 'CSS' on the right with a code editor. At the bottom, there's a 'JavaScript + No-Library (pure JS)' panel and a 'Result' panel.

← → ↻ jsfiddle.net

Close

Start with a boilerplate:

jQuery Vue React React + JSX Preact TypeScript

CoffeeScript SCSS CSS Grid (responsive) Bootstrap

PostCSS New

Show boilerplate bar less often?

Links:

- 👍 Roadmap (vote for features)
- 🐛 Bug tracker
- ☁ About
- 📄 Docs
- 🏠 Service status

JSFiddle is for:

- Demos for docs
- Bug reporting (test-case) for Github Issues
- Presenting code answers on Stack Overflow
- Live code collaboration
- Code snippets hosting

... or just your humble code playground 🙌

▶ Run ↺ Save ≡ Tidy 💬 Collaborate Update New and updated JS/CSS linters ⚙ Settings Sign in

**Fiddle meta**

Untitled fiddle

No description

Add title to make the fiddle visible on your profile page

Resources URL cdnjs

Async requests

Other (links, license)

HTML ▼

1

CSS ▼

1

JavaScript + No-Library (pure JS) ▼

1

Result

# Online Editor – JSFiddle(<https://jsfiddle.net>) (Cont.)

The screenshot displays the JSFiddle online editor interface. The browser's address bar shows the URL `jsfiddle.net/9wuma8by/`. The interface includes a top navigation bar with options like 'Run', 'Save', 'Tidy', 'Collaborate', 'Update', and 'Settings'. On the left, the 'Fiddle meta' panel shows 'Untitled fiddle' and 'No description'. The main editor area is divided into three sections: 'HTML', 'CSS', and 'JavaScript + No-Library (pure JS)'. The 'HTML' section contains a script tag and a div with id 'app'. The 'CSS' section is empty. The 'JavaScript' section contains a Vue.js instance with a template and data. The preview area on the right shows the output 'Hello, World!'. Two green circles with numbers '1' and '2' are overlaid on the 'Fiddle meta' and 'HTML' sections respectively. A 'Headway' logo is visible in the bottom left corner.

Create a new fiddle - JSFiddle

jsfiddle.net/9wuma8by/

Run Save Tidy Collaborate Update New and updated JS/CSS linters Settings Sign in

**Fiddle meta**

1

2

Untitled fiddle

No description

Add title to make the fiddle visible on your profile page

Resources URL cdnjs

Async requests

Other (links, license)

Headway

Easiest way to keep your customers

HTML

```
1 <script src="https://unpkg.com/vue"></script>
2
3 <div id="app">
4
5 </div>
```

CSS

```
1
```

JavaScript + No-Library (pure JS)

```
1 new Vue({
2   template : '<p>{{msg}}</p>',
3   data : {
4     msg : 'Hello, World!'
5   }
6 }).$mount('#app')
```

Hello, World!

# Online Editor – CodePen(<https://codepen.io>)

The screenshot shows the CodePen website homepage. The browser's address bar displays 'codepen.io'. The page features a dark theme with a prominent white CodePen logo and the headline 'The best place to build, test, and discover front-end code.' Below this, a description states: 'CodePen is a social development environment for front-end designers and developers. Build and deploy a website, show off your work, build test cases to learn and debug, and find inspiration.' A green 'Sign Up for Free' button is positioned below the text. To the right, three overlapping code snippets are shown: HTML, SCSS, and JS. The HTML snippet contains a simple div with a class. The SCSS snippet defines a linear gradient for a class named '.rect'. The JS snippet defines an array of colors and an 'animate' function. The left sidebar contains navigation links such as 'Start Coding', 'EXPLORE', 'Picks', 'Popular Topics', 'GROW', 'Jobs', 'CodePen PRO', 'Challenges', 'COMPANIES', 'Hire Talent', 'Create a Team', 'Advertise', 'Newsletter', 'Blog', and 'Docs & Support'. At the bottom, there are three main sections: 'Build & Test', 'Learn & Discover', and 'Share Your Work'. An 'Adobe Stock' advertisement is visible in the bottom left corner, and a Windows activation watermark is in the bottom right.

CodePen: Build, Test, and Discover x +

codepen.io

CODEPEN

TRY OUR ONLINE EDITOR

Start Coding

EXPLORE

Picks

Popular Topics

GROW

Jobs

CodePen PRO

Challenges

COMPANIES

Hire Talent

Create a Team

Advertise

Newsletter

Blog

Docs & Support

MAKE IT WITH ADOBE STOCK. Get 10 free images >

Limited time offer: Get 10 free Adobe Stock images.

Search CodePen...

Sign Up

Log In

The best place to build, test, and discover front-end code.

CodePen is a **social development environment** for front-end designers and developers. Build and deploy a website, show off your work, build test cases to learn and debug, and find inspiration.

Sign Up for Free

HTML

```
<div class="rect"></div>
```

SCSS

```
.rect {  
  background: linear-gradient(  
    -119deg,  
    $gray 0%,  
    $dark-gray 100%);  
}
```

JS

```
var colors =  
["#74B087", "#DE7300", "#74B087"];  
  
function animate() {};
```

Build & Test

Learn & Discover

Share Your Work

Activate Windows  
Go to Settings to activate Windows.

# Online Editor – CodeSandbox(<https://codesandbox.io>)

