

LIVE

จัดอบรมออนไลน์

Advanced Nuxt 3

ขั้นสูง

518
วัน
ซั่วโมงเต็ม

21,28 Jan , 4,11,18 Feb 2023

20.00-23.30 PM



สอนสดผ่าน Zoom
รับจำนวนจำกัด



อาจารย์สามิตร
สถาบันไอทีจีเนียล



มีวิดีโอบันทึกอบรม
ย้อนหลังให้ทุกวัน



สามิตร ໂກຍນ

ปรญญาโทคณะเทคโนโลยีสารสนเทศ

มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ

▶ Frontend

Angular, React, NextJS, VueJS,
Bootstrap, TailwindCSS

▶ Backend

PHP, Python, Java, NodeJS, ASP.net

▶ Database

MySQL, PostgreSQL, MS SQL Server,
MongoDB

▶ Mobile

Java, Kotlin, Objective C, Swift, Cordova,
ionic, Flutter

▶ DevOps

Git, Github, Gitlab, Docker,
Kubernetes

Course Outline



- Section 1: ปรับโครงสร้างโปรเจ็คต์ Nuxt 3 รองรับงานขนาดใหญ่ขึ้น
- Section 2: การจัดการข้อผิดพลาดฝั่ง Client และฝั่ง Server
- Section 3: การปรับปรุงฟังก์ชันด้วยการใช้ Utilities
- Section 4: การเขียนเว็บแอปหลายภาษาด้วย Nuxt 3 i18n
- Section 5: การสร้าง API ด้วย Nuxt 3
- Section 6: Nuxt 3 Authentication and Middleware
- Section 7: Nuxt 3 JWT authentication
- Section 8: Nuxt 3 with Data Fetching and HTTP Requests
- Section 9: Nuxt 3 with Prisma in Postgres database
- Section 10: Nuxt 3 State management with Pinia
- Section 11: การ Build และ Deployed project ไปใช้งานจริง



ดาวน์โหลดเอกสารประกอบการอบรม
bit.ly/nuxt3advanced

LIVE

จัดอบรมออนไลน์

Advanced Nuxt 3

ขั้นสูง

5 วัน
18 ชั่วโมงเต็ม

🗓 21,28 Jan , 4,11,18 Feb 2023

⌚ 20.00-23.30 PM



zoom

สอนสดผ่าน Zoom
รับจำนวนจำกัด



มีวิดีโอบันทึกอบรม
ย้อนหลังให้ทุกวัน

วันที่ 1

LIVE

จัดอบรมออนไลน์

Advanced Nuxt3

21,28 Jan , 4,11,18 Feb 2023

20.00-23.30 PM



สอนสดผ่าน Zoom
รับจำนวนจำกัด



การเตรียม เครื่องมือ



อาจารย์สามิต
สถาบันไอทีเนียส



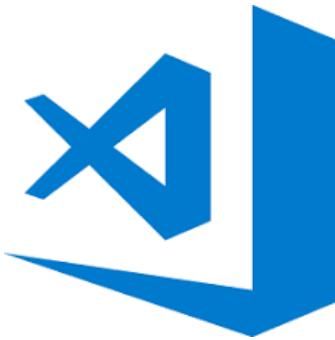
มีวิดีโอบันทึกอบรม
ย้อนหลังให้ทุกวัน

โปรแกรม (Tool and Editor) ที่ใช้wbrm

- 1. Visual Studio Code + extension**
- 2. Node.JS 16+**
- 3. Git**

แนะนำ หลักสูตรนี้ใช้ Node.JS เวอร์ชัน 16 ขึ้นไป

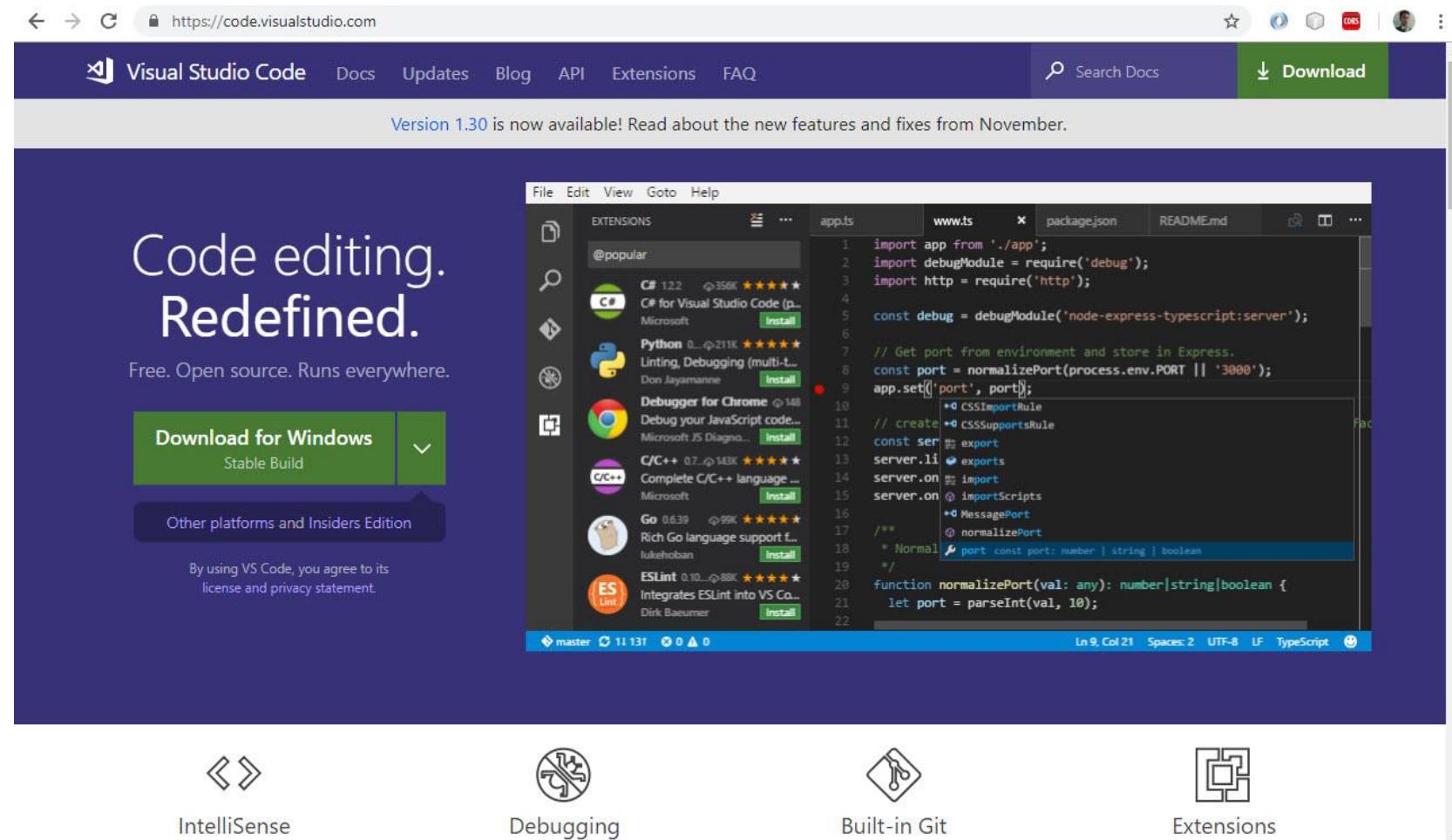




ติดตั้ง Visual Studio Code



ติดตั้ง Visual Studio Code พร้อมส่วนเสริมที่จำเป็น



IntelliSense



Debugging



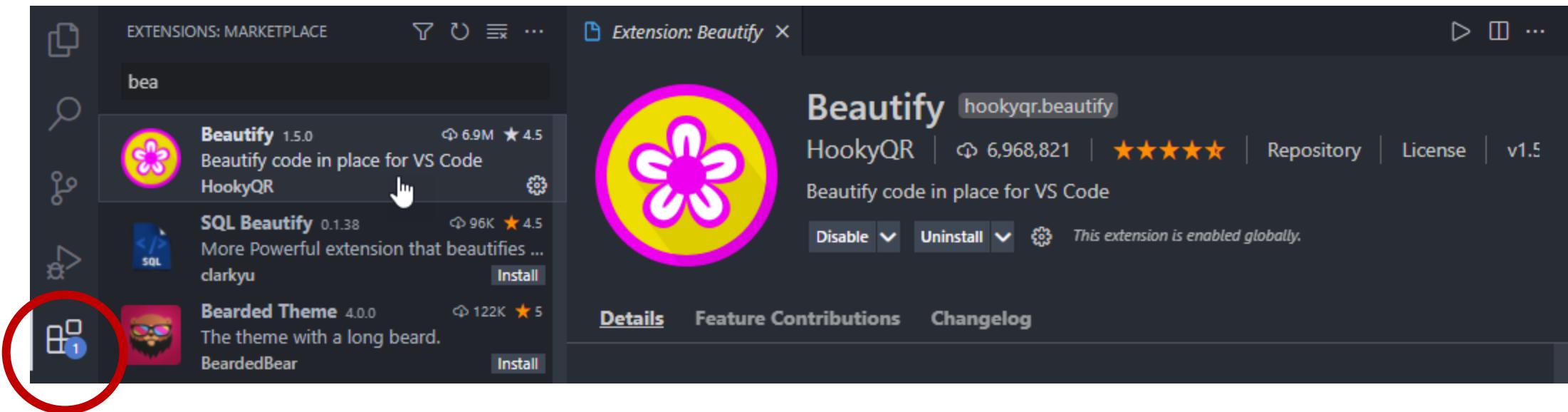
Built-in Git



Extensions

เข้าไปดาวน์โหลด Visual Studio Code ได้ที่ <https://code.visualstudio.com>

การติดตั้งส่วนเสริม (Extension) ของ Visual Studio Code



รายการ Extension ที่แนะนำสำหรับ Nuxt 3

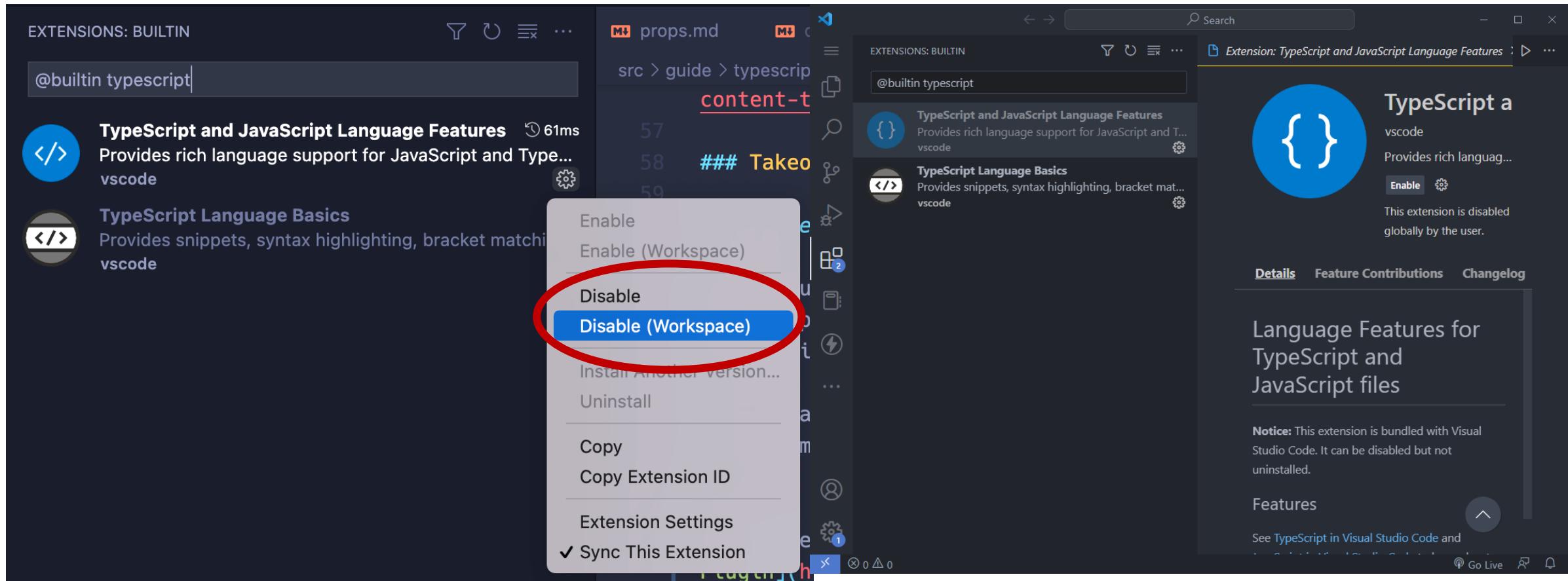
Extension พื้นฐาน

1. **Color Picker** by anseki
2. **Color Highlight** by Sergii N
3. **AutoFileName** by JerryHong
4. **Auto Import – ES6, TS, JSX, TSX** by Sergey Korenuk
5. **Highlight Matching Tag** by vincaslt
6. **Material Icon Theme** by Philipp Kief
7. **Prettier – Code formatter** by Prettier
8. **Rainbow Brackets** by 2gua
9. **Thunder Client** by Ranga Vadhineni
10. **One Dark Pro** by binaryify

Extension for Vue / Nuxt

1. **Vue Language Features (Volar)** by Vue
2. **TypeScript Vue Plugin (Volar)** by Vue
3. **Vue VSCode Snippets** by sarah.drasner
4. **Vue 3 Snippets** by hollowtree
5. **Tailwind CSS IntelliSense** by Tailwind Labs
6. **vuetify-vscode** by Vuetifyjs

อย่าลืม ! Disable @builtin typescript



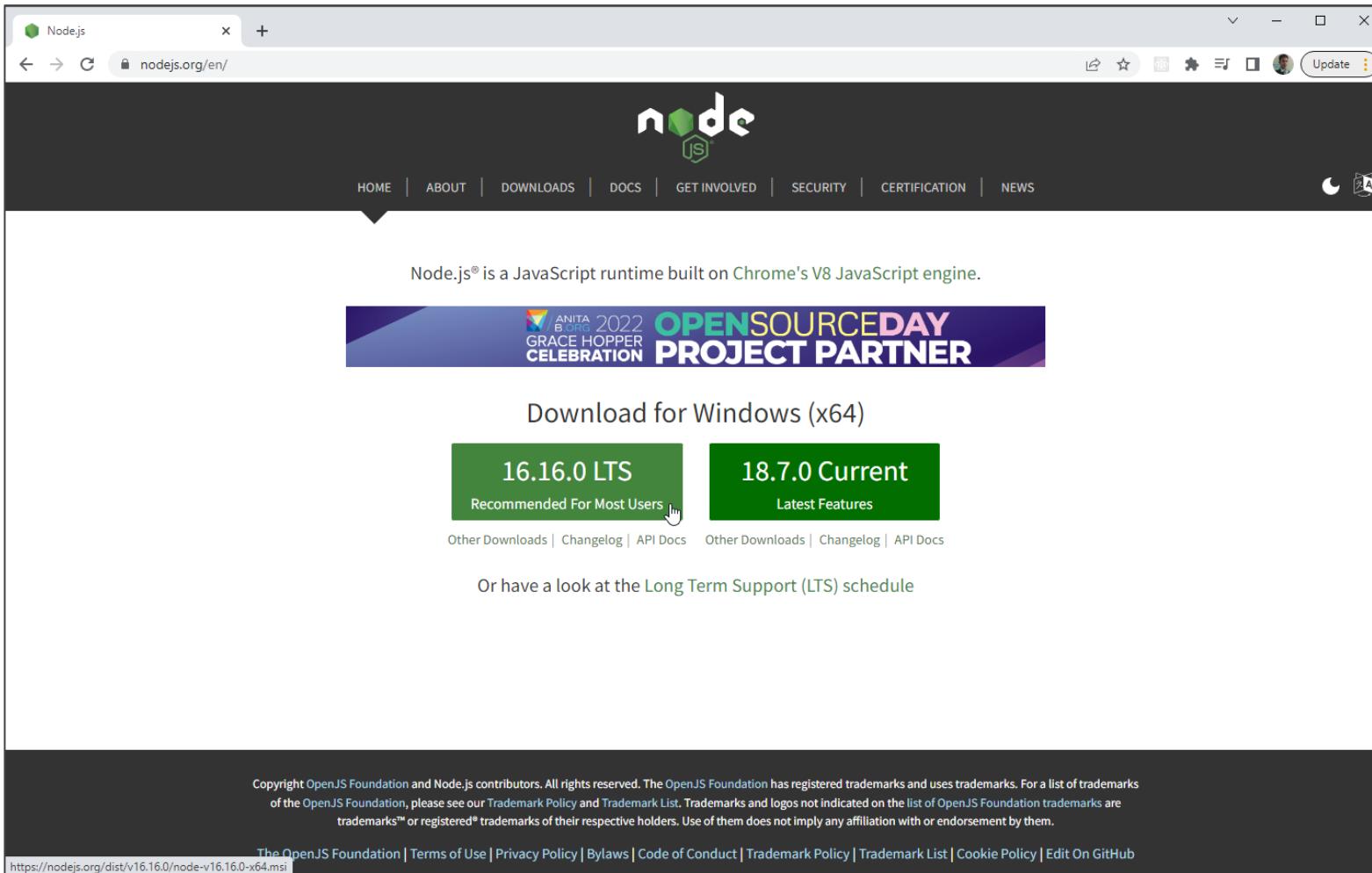


ติดตั้ง Node JS 16+

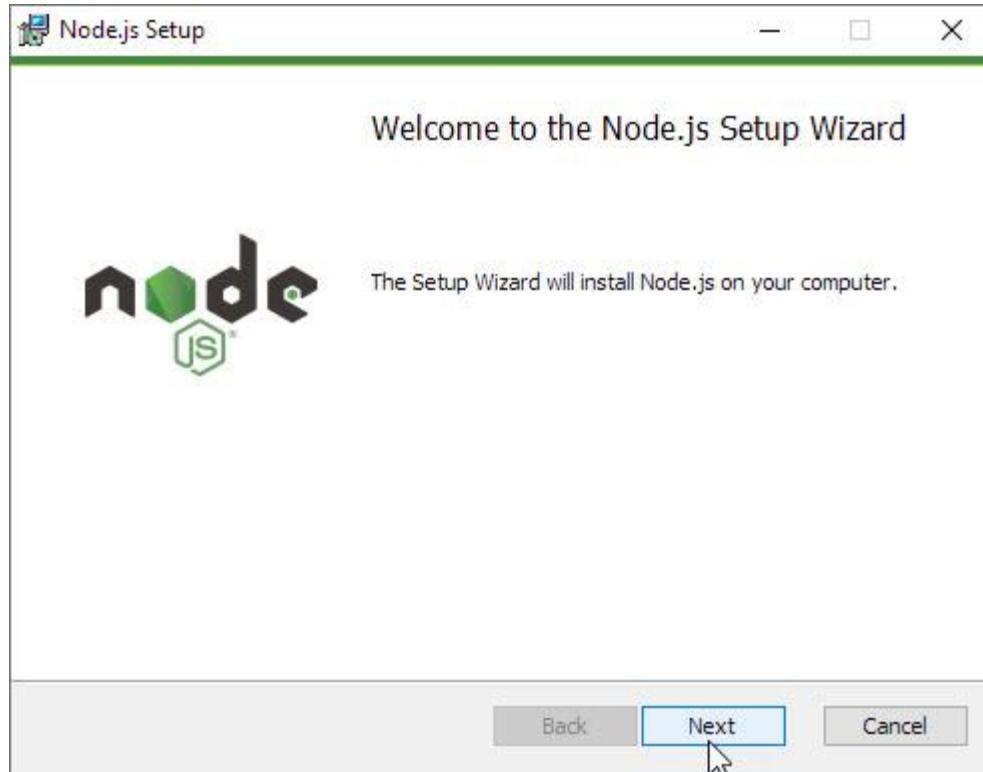


Download Node.js V.16.16.x

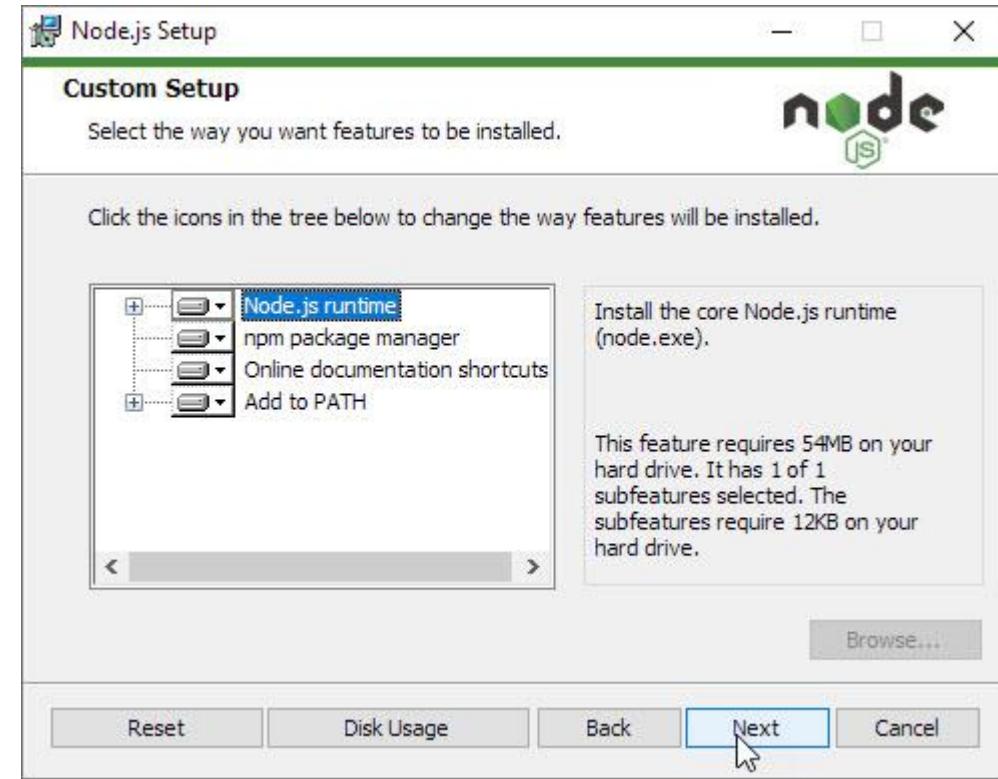
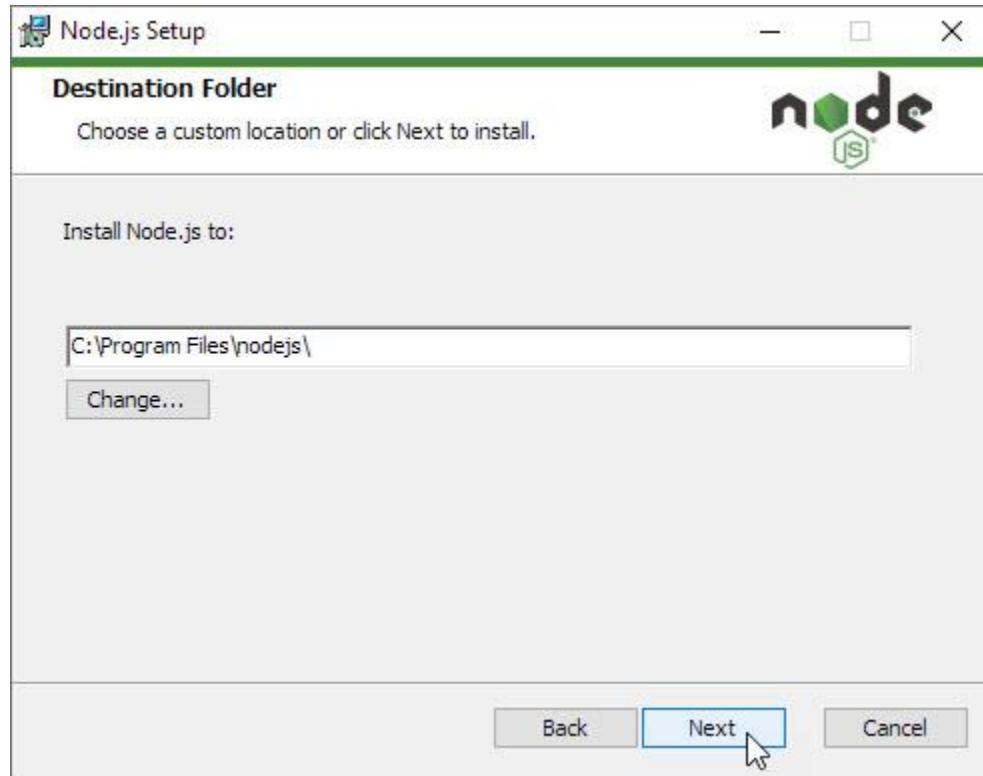
<https://nodejs.org/en/>



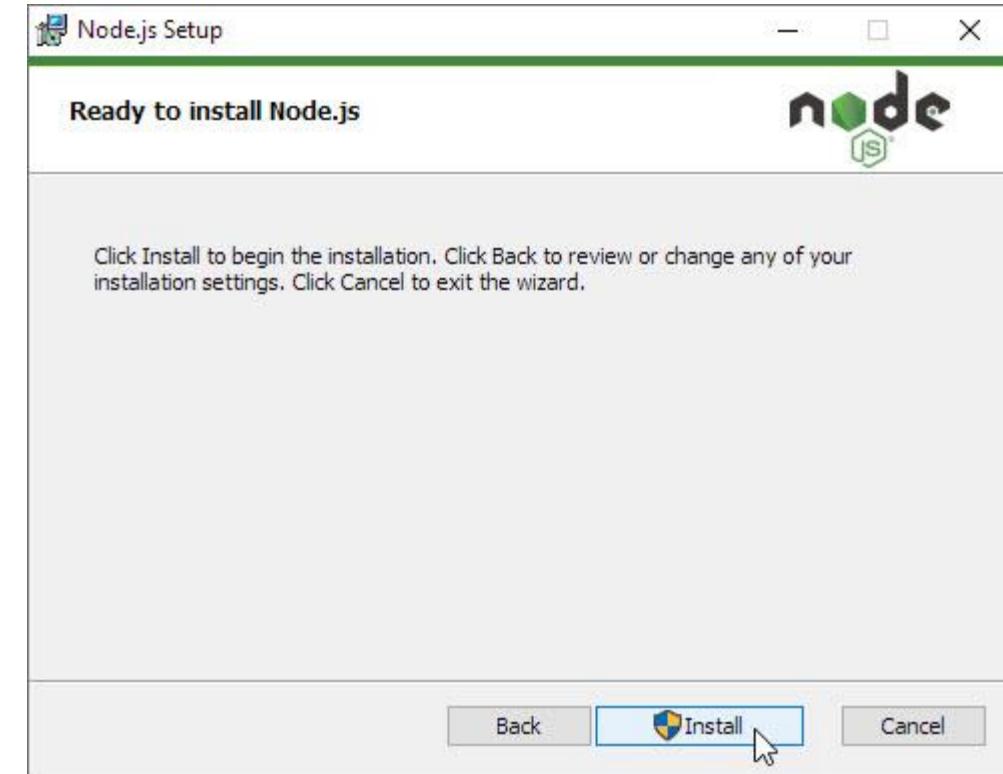
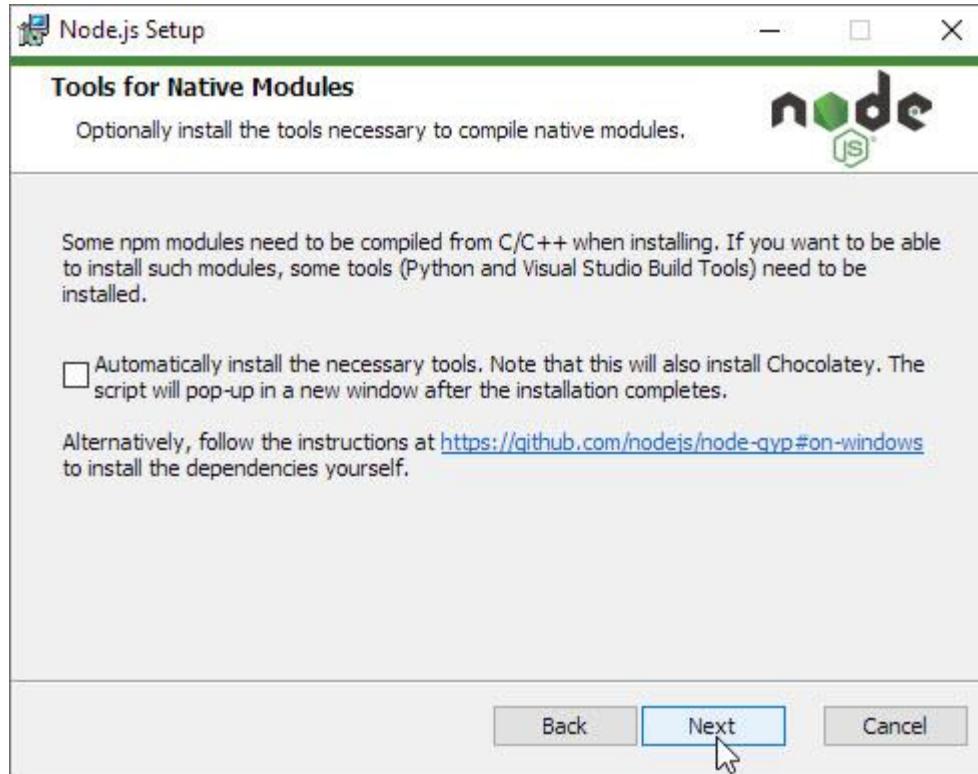
ติดตั้ง Node.js 16.16.x



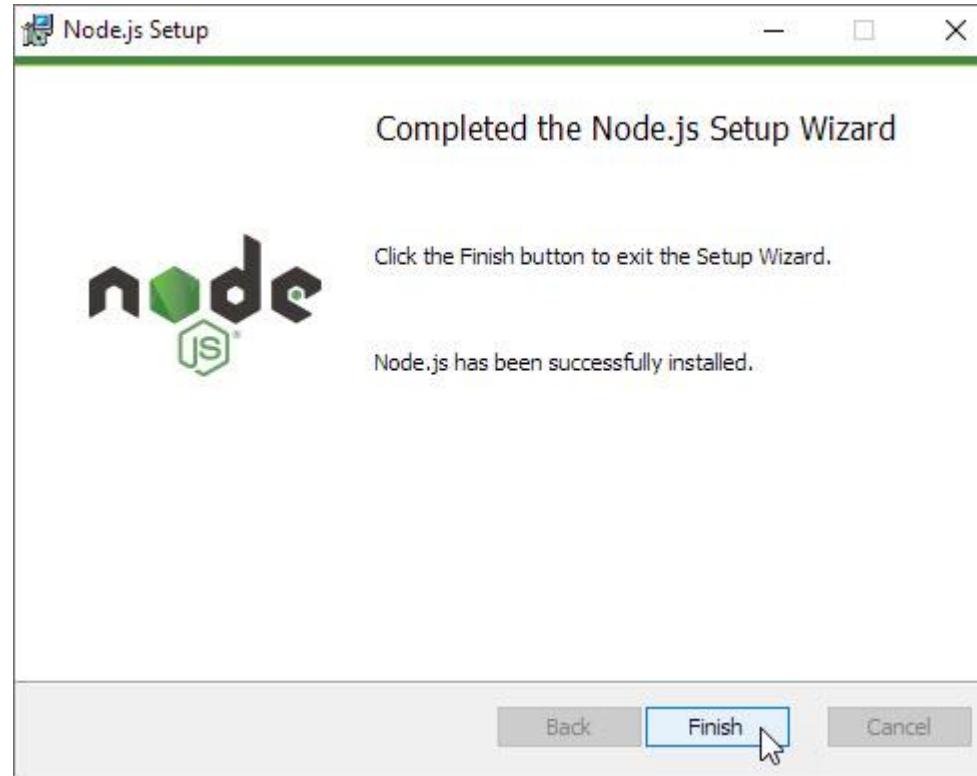
ติดตั้ง Node.js 16.16.x



ติดตั้ง Node.js 16.16.x

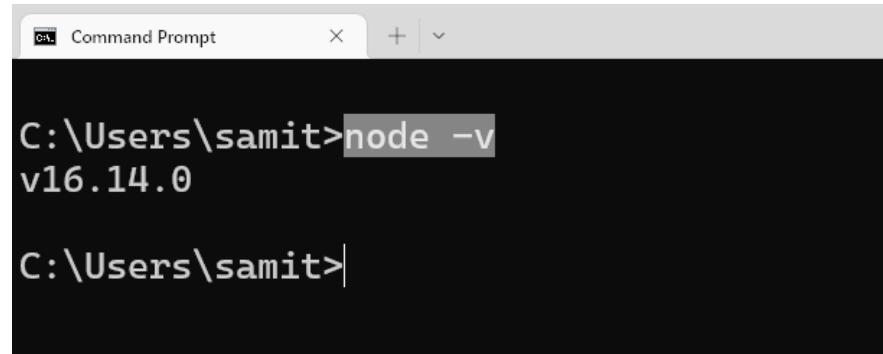


ติดตั้ง Node.JS 16.16.x



ทดสอบชุดติดตั้งเสร็จ

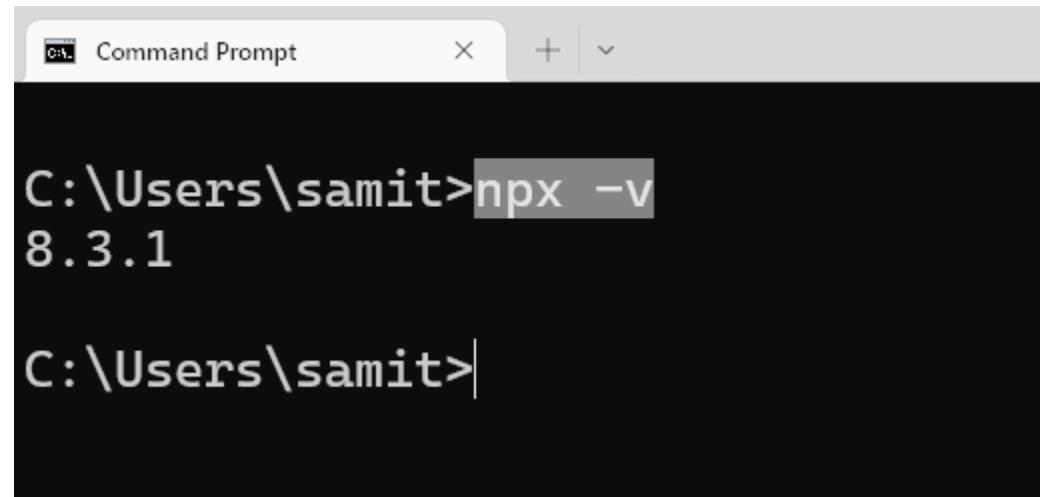
node -v



```
C:\Users\samit>node -v
v16.14.0

C:\Users\samit>
```

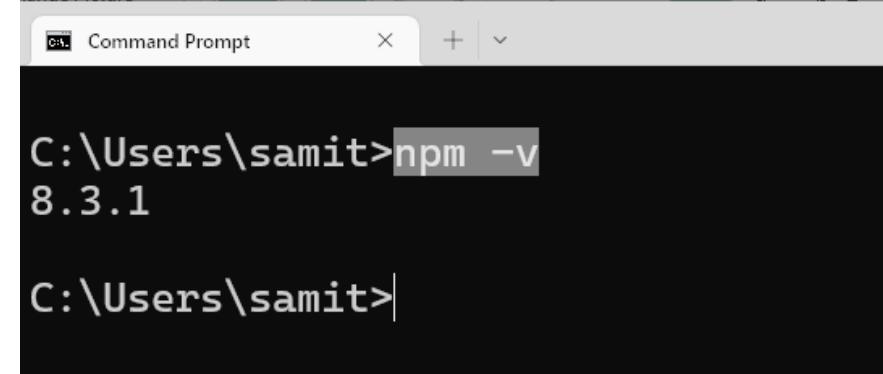
npx -v



```
C:\Users\samit>npx -v
8.3.1

C:\Users\samit>
```

npm -v



```
C:\Users\samit>npm -v
8.3.1

C:\Users\samit>
```



git --fast-version-control

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.

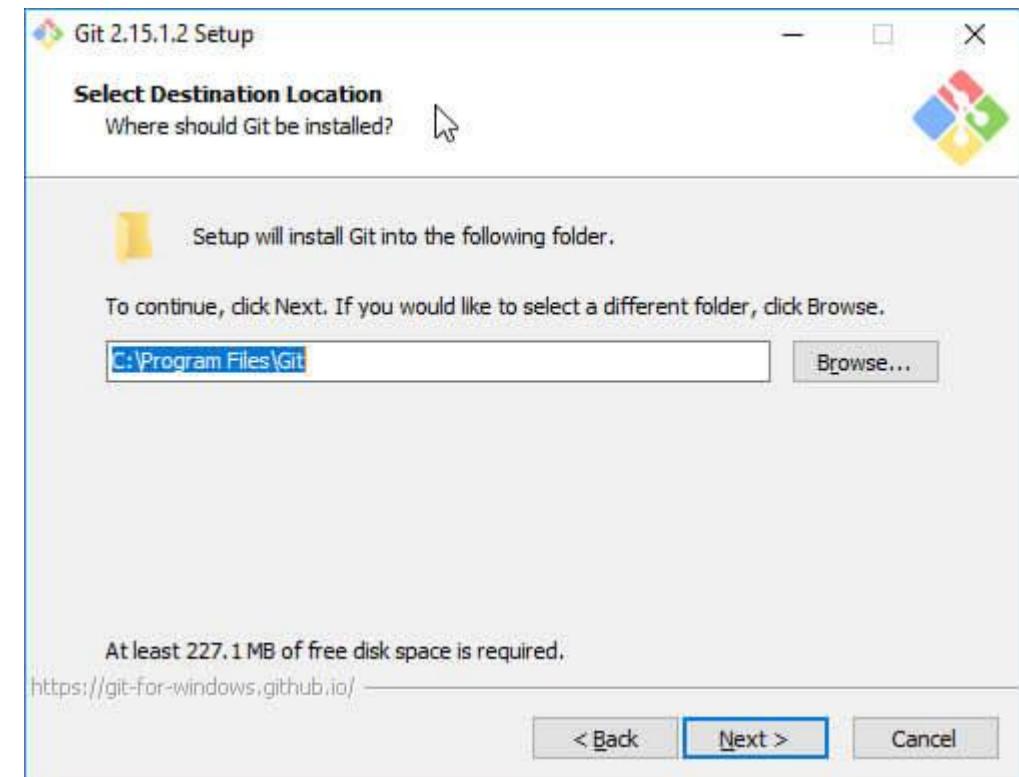
Downloads
GUI clients and binary releases for all major platforms.

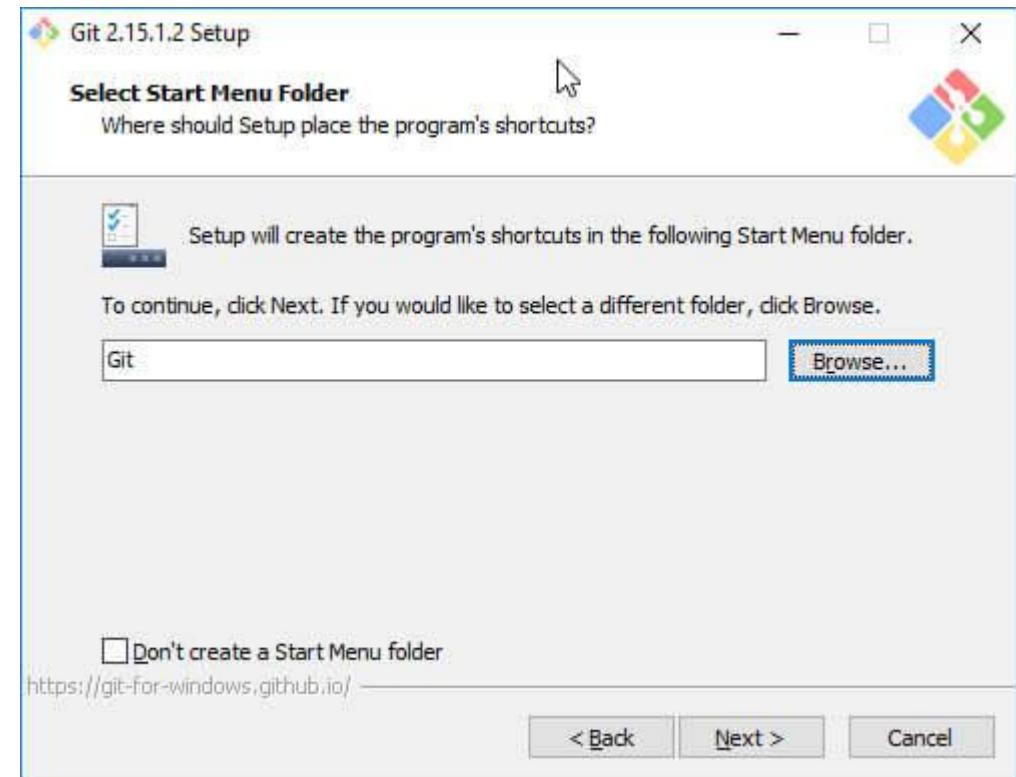
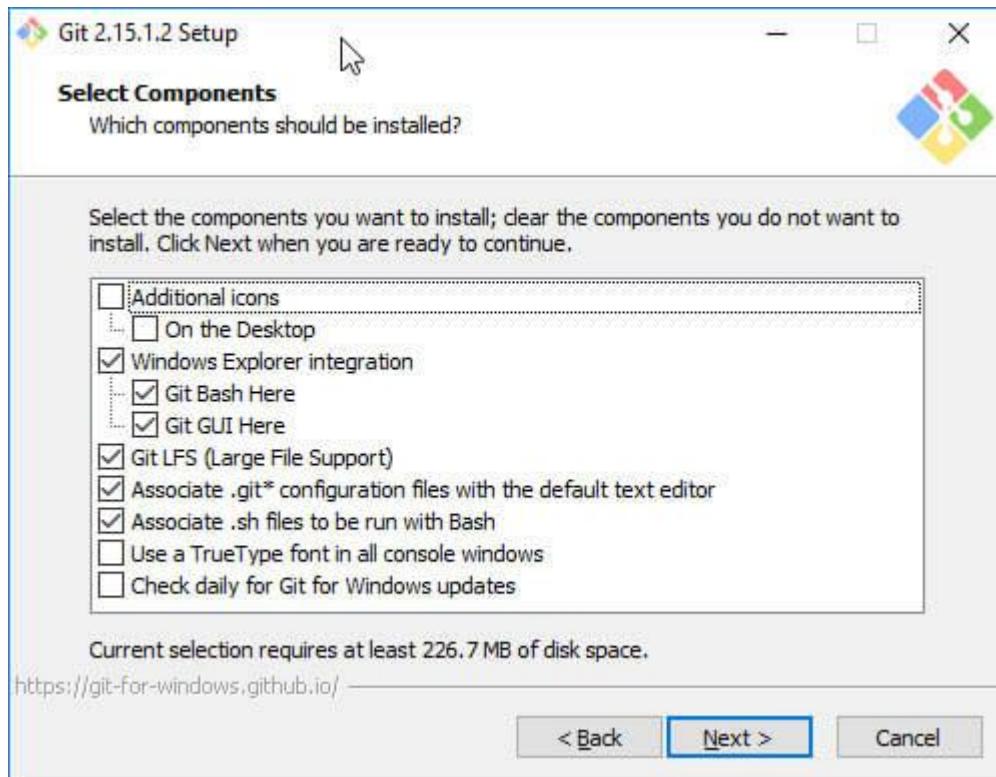
Documentation
Command reference pages, Pro Git book content, videos and other material.

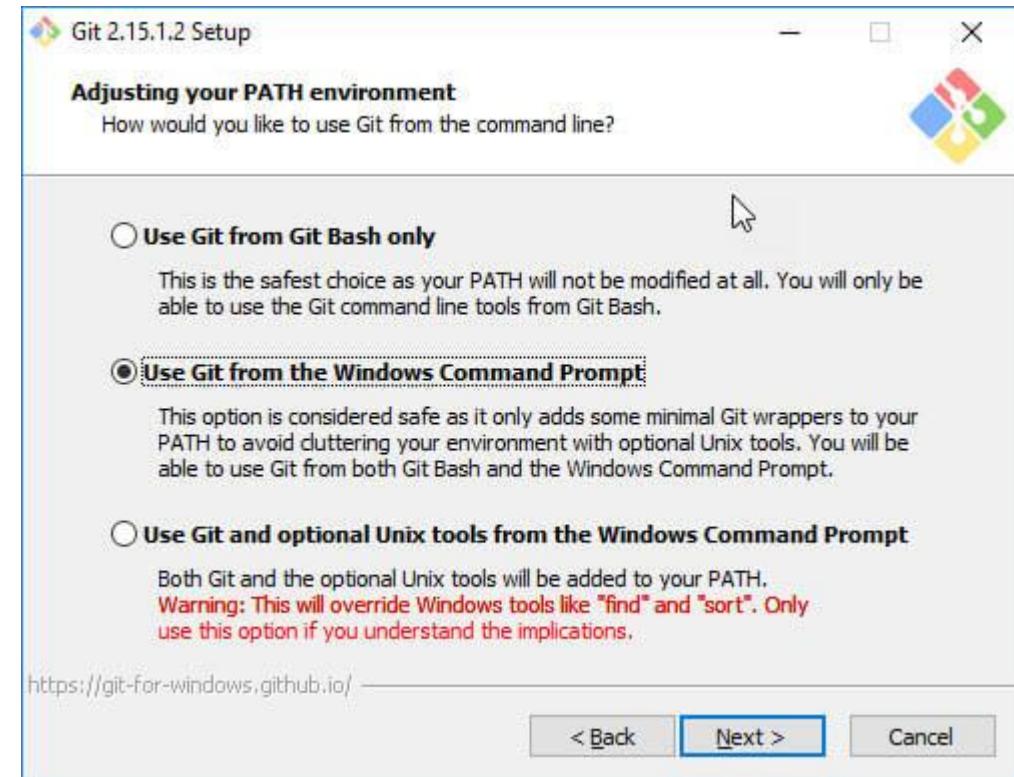
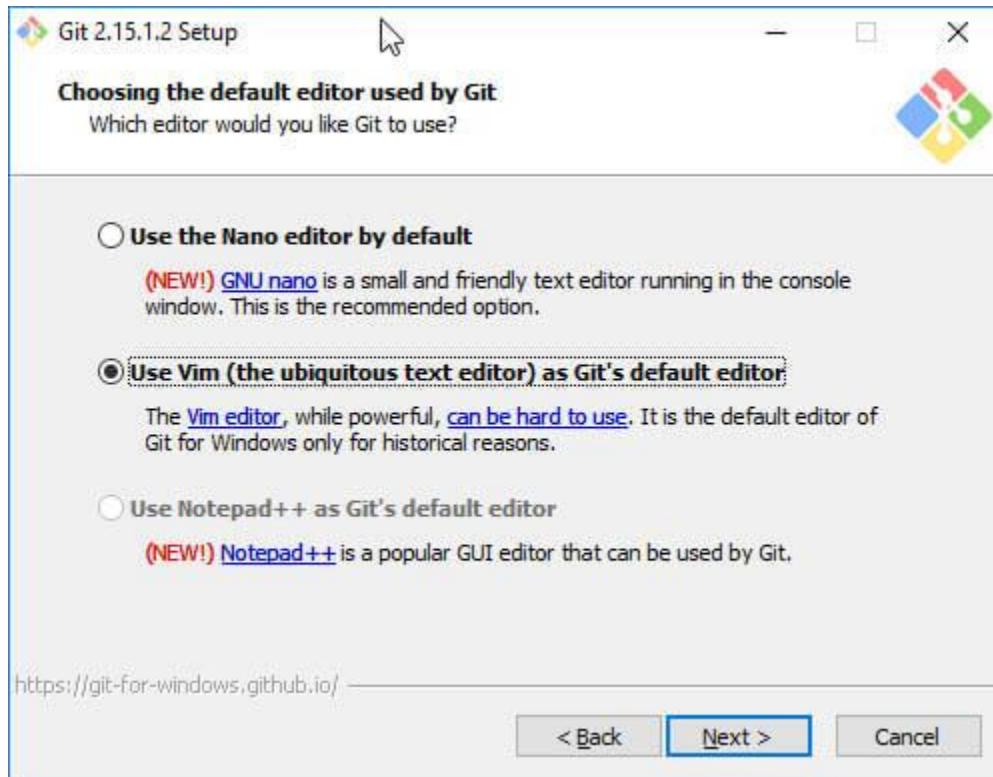
Community
Get involved! Bug reporting, mailing list, chat, development and more.

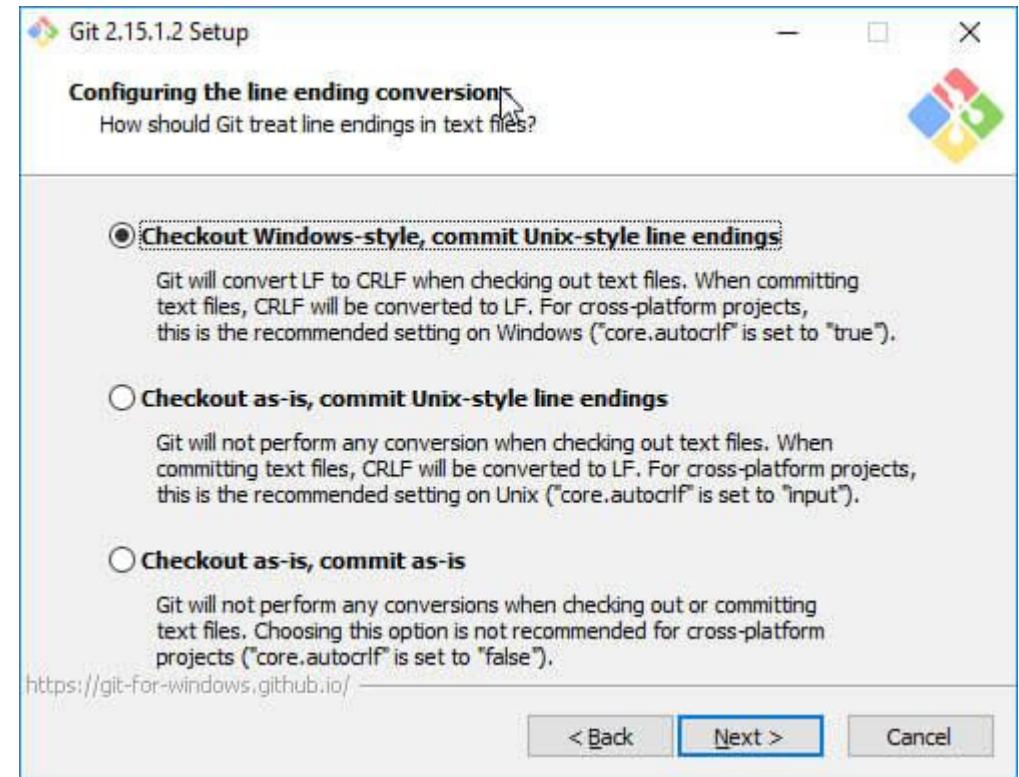
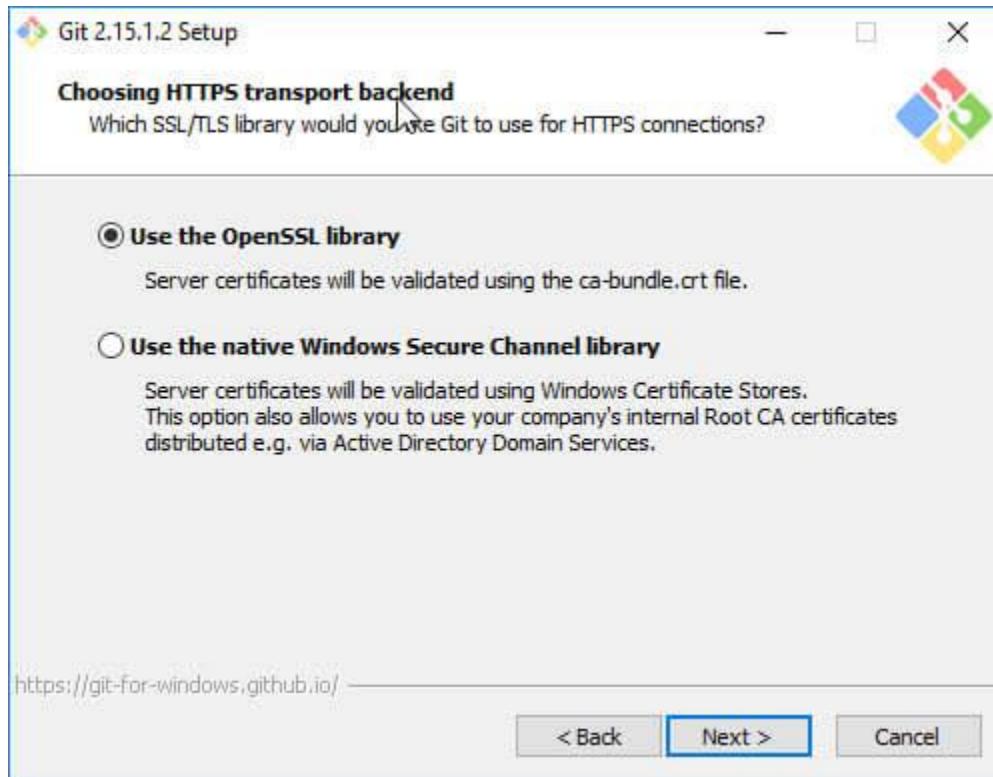
Latest source Release
2.22.0
Release Notes (2019-06-07)
Download 2.22.0 for Windows

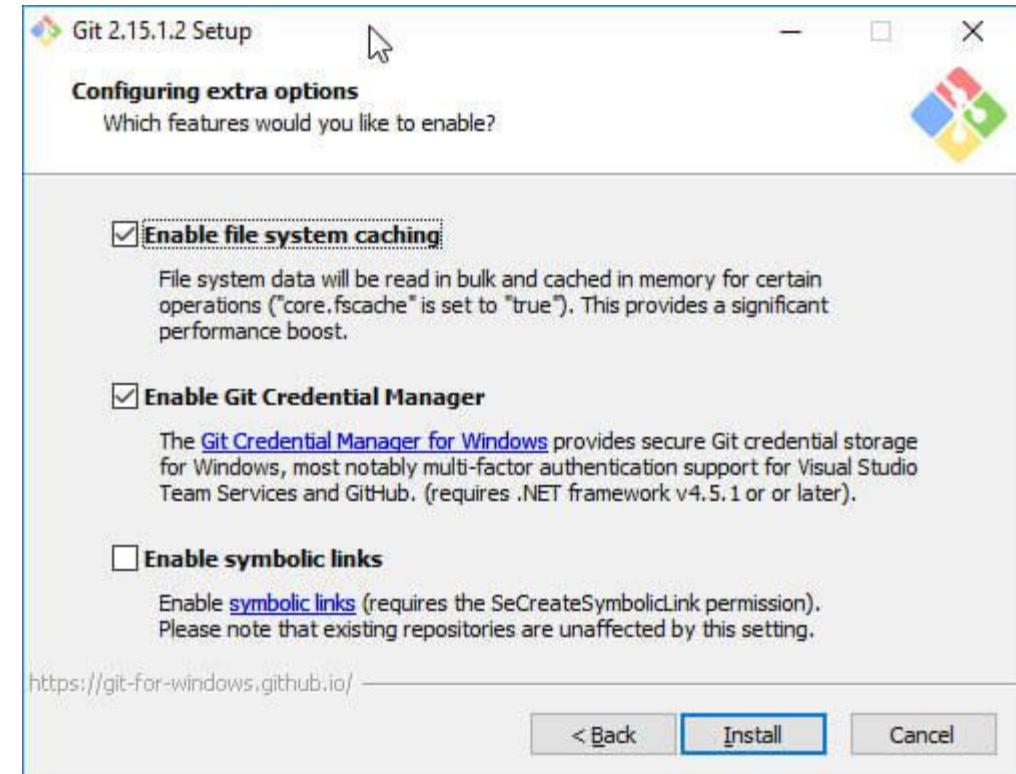
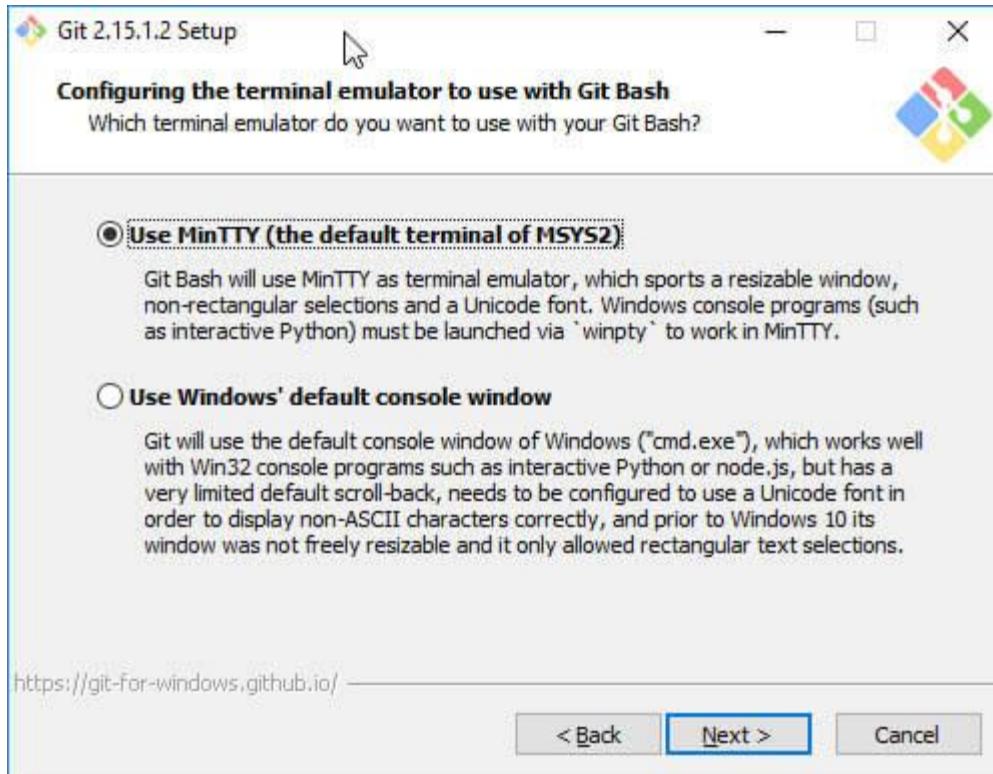
<https://git-scm.com/download/win>











C:\Windows\system32\cmd.exe

```
C:\Users\samit>git --version  
git version 2.21.0.windows.1  
C:\Users\samit>
```

MINGW64:/c/Users/samit

```
samit@DESKTOP-5KFFN0F MINGW64 ~  
$
```

git --version



The Intuitive Web Framework

3.0 stable

nuxt.com



Presented by
 Vue
School.io

<https://vueschool.io/articles/news/nuxt-3-stable-launch-all-the-details-from-nuxt-nation-2022/>

What's different in Nuxt 3

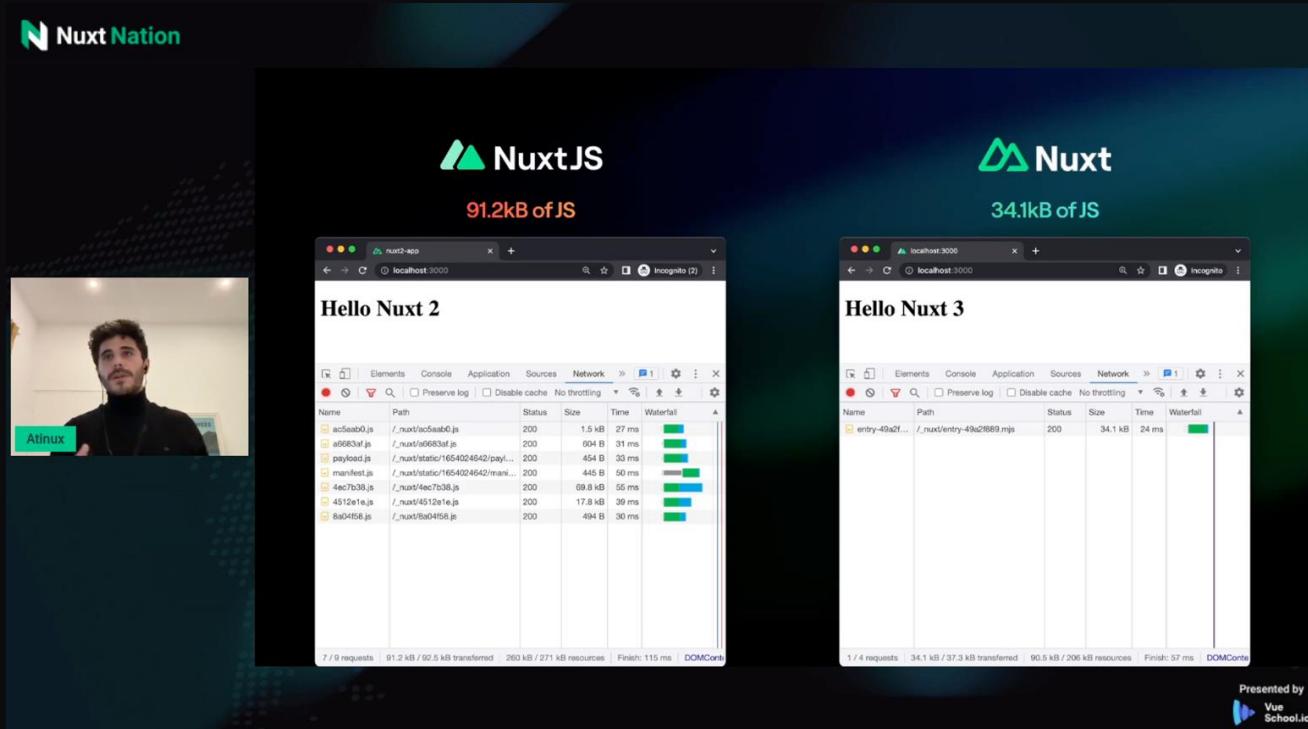
The slide features a dark background with a green circular logo in the top left corner containing the text "Nuxt Nation". In the center, there is a large white text "Today". To the right, there are two logos: "NuxtJS" with a green triangle icon and "Nuxt" with a blue triangle icon. Below each logo is its respective website address: "nuxtjs.org" and "nuxt.com". A comparison table follows:

	NuxtJS	Nuxt
UI Framework	Vue 2	Vue 3
Web Server	connect	h3
Bundler	Webpack 4	Vite / Webpack 5
Routing Library	Vue Router 3	Vue Router 4
Meta Tags Manager	Vue Meta	VueUse Head
Server(less) Packager	-	Nitro

At the bottom right, it says "Presented by" with the "Vue School.io" logo.

- Nuxt 3 uses Vue 3. This provides out of the box support for the composition API, better performance, and more
- Nuxt 3 uses a h3 web server which is an upgrade from the connect web server in Nuxt 2. H3 is a minimal framework built on top of unjs for high performance and can work in multiple environments.
- Nuxt 3 uses Vite for bundling and can also support webpack 5 via a plugin
- Uses Vue-router 4
- Uses composable VueUse Head to handle meta data and SEO
- A Server(less) packager Nitro
- Much smaller and lighter Bundle size

What's different in Nuxt 3



- App Entry with Hydration
- Root component with ``
- Universal and lightweight router `useRouter()`, `middleware`, and ``
- Head composable and components like `useHead()`, `<Title>`, , and for SEO and social sharing
- Universal data fetching with `$fetch()`
- Default 400 and 500 error pages
- Plugins and `runtimeConfig` logic
- `useNuxtApp()` composable and hooks `app:created`, `app:mounted`, `page:start` for hooking into Nuxt lifecycle. Useful for creating transitions and loading animations.

What's different in Nuxt 3



Fast and Furious

Optimized with code-splitting, tree-shaking, optimized cold-start, link prefetching, payload extraction, just to name a few. Fast by default so you can focus on building.



Hybrid Rendering

Decide what rendering strategy at the route level: SSR, SSG, CSR, ISR, ESR, SWR. Build any kind of website or web application with best performance in mind.



SEO and Web Vitals

By leveraging server-side rendering, ESM format and optimized images. Nuxt websites are indexable by search engines while giving an application feeling to the end-users.

Section 1

ปรับโครงสร้างโปรเจ็คต์ Nuxt 3 รองรับ งานขนาดใหญ่ขึ้น

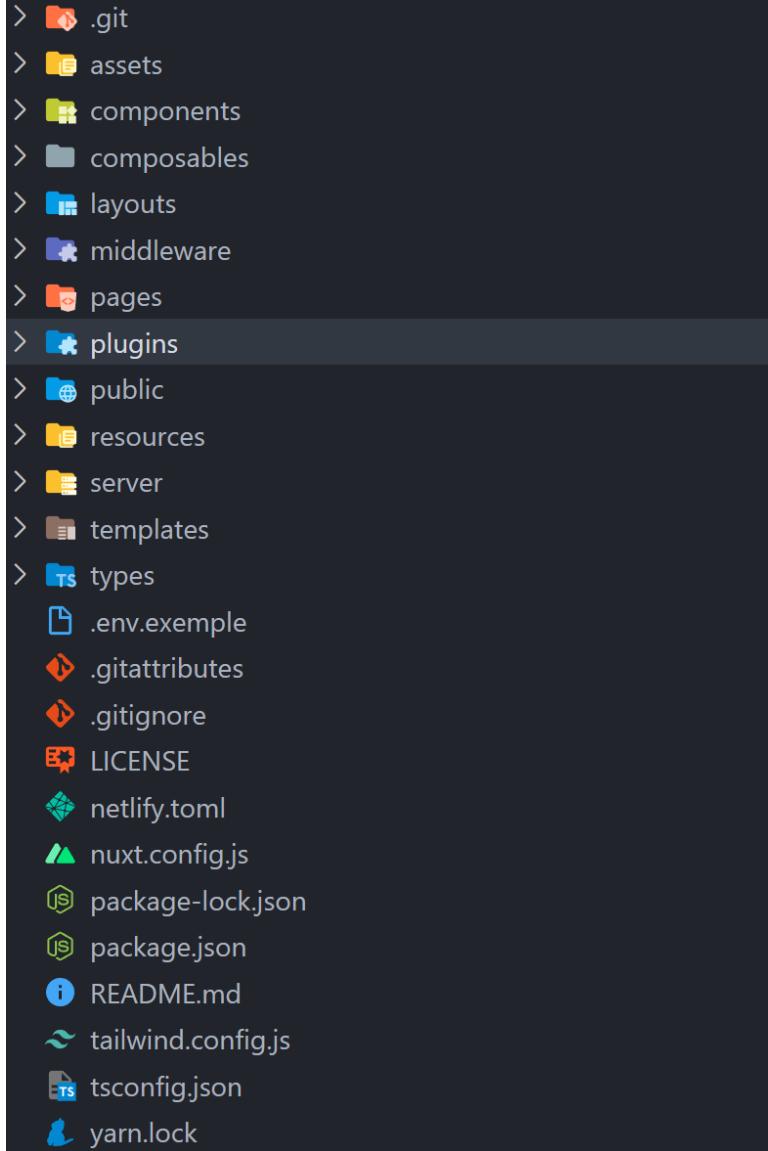
- แนะนำตัวอย่างโครงสร้างโปรเจ็คต์ที่นิยมใช้ในการ Scaleable ได้ในอนาคต
- การสร้าง Layout แบบ Custom
- การเขียนฟังก์ชันการทำงานแยกไว้ใน Composables
- การเรียกใช้งาน Tailwind CSS, Vuetify , Fonts, Icons และอื่นๆ



Nuxt 3 Scalable Project Structure

Understanding the
Directory
Structure in Nuxt 3





assets

The assets folder holds all website's assets the build tools will process. Usually your stylesheets, fonts and images that are not going to be served by the server can be placed in this folder.

components

Components is one folder we are all familiar with. It holds all our Vue components to be reused in our pages or other components. Nuxt 3 auto imports all our components created in this folder so we don't have to manually import them before using them.

Composables

Composables are functions that leverage Vue.js composition API to create reusable stateful logic. Composables help in preventing repetition in writing functions that apply in multiple components. Just like components, Nuxt 3 auto imports composites created in this folder.

Layouts

Layouts, as the name implies, provides a structure for your Vue.js pages. As your app complexity increases you may want to define different layouts for different pages or components. The layout folder holds your layout files and are automatically loaded via asynchronous import. Layouts are used by adding to your app.vue, and either setting a layout property as part of your page metadata.

- > .git
- > assets
- > components
- > composable
- > layouts
- > middleware
- > pages
- > plugins
- > public
- > resources
- > server
- > templates
- > types
 - .env.example
 - .gitattributes
 - .gitignore
 - LICENSE
 - netlify.toml
 - nuxt.config.js
 - package-lock.json
 - package.json
 - README.md
 - tailwind.config.js
 - tsconfig.json
 - yarn.lock

Middleware

With Nuxt 3, you can include a customizable route middleware framework throughout your application, allowing you to extract code you want to run before traversing a particular route. This middleware only runs on the client side which is the Vue.js part of your Nuxt application

Pages

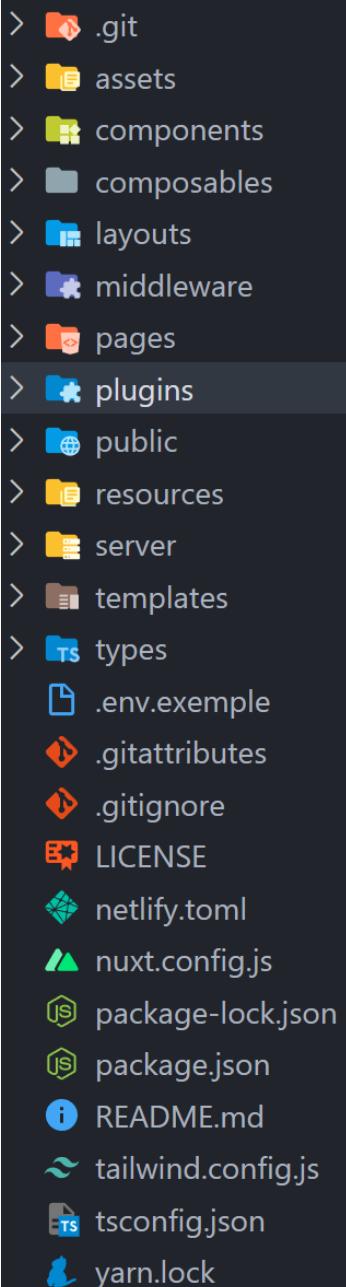
Pages is where all our routes are defined. This is one exciting feature in Nuxt 3 that I am always pumped about. Vue router has been installed under-the-hood so creating routes for our nuxt application has become so simple. Vue-router is optional as to activate it you will need to create your pages directory

Plugins

Plugins *are self-contained code that usually add app-level functionality to your nuxt app.* Nuxt automatically reads the files in your `plugins` directory and loads them at the creation of the Vue application. All plugins in your `plugins/` directory are auto-registered, so you should not add them to your `nuxt.config` separately. Just like `composables` plugins on the top level of the `plugins` directory can be registered and also index files in subdirectories

Public

This was previously known as the static folder in Nuxt 2 but in Nuxt 3 the `public/` directory is directly served at the server root and contains public files that have to keep their names (e.g. `robots.txt`) or likely won't change (e.g. `favicon.ico`)



resources

For some custom resource template like a mail template etc.

Server

The server folder contains all files needed on the server side of your Nuxt application. This includes apis , server middleware, server routes, server plugins and server utilities

templates

For custom root html template of nuxt framework

types

For keep type of typescript declarations

.env.example

For keep environment of project

.nuxt.config.js

Configuration file of nuxt framework

package.json

Node module package dependencies define

tailwind.config.js

Configuration file of tailwind css framework

Tsconfig.json

Typescript configuration file

Nuxt 3

Create Nuxt Project



Nuxi (Nuxt CLI)



nuxi add

```
npx nuxi add [--cwd] [--force] <TEMPLATE> <NAME>
```

Option	Default	Description
TEMPLATE	-	Specify a template of the file to be generated.
NAME	-	Specify a name of the file that will be created.
-- cwd	.	The directory of the target application.
-- force	false	Force override file if it already exists.

Create a Nuxt 3 Project

with npx or pnpm

npx pnpm

```
> npx nuxi init <project-name>
```

npx

npx pnpm

```
> pnpm dlx nuxi init <project-name>
```

pnpm

Create a Nuxt 3 Project

Install the dependencies

```
yarn   npm   pnpm  
➤ npm install          npm
```

```
yarn   npm   pnpm  
➤ yarn install          yarn
```

```
yarn   npm   pnpm  
➤ pnpm install          pnpm
```

Exercise Create a Nuxt 3 Project

```
$ mkdir Nuxt3Advanced
```

สร้างโฟลเดอร์ใหม่

```
$ cd Nuxt3Advanced
```

เปลี่ยน path ไปยังโฟลเดอร์ที่สร้าง

```
$ npx nuxi init nuxt3-scalable-tailwind
```

คำสั่งสร้างโปรเจกต์ใหม่ใน Nuxt 3

```
$ code nuxt3-scalable-tailwind -r
```

เปิดโปรเจกต์เข้า VSCode

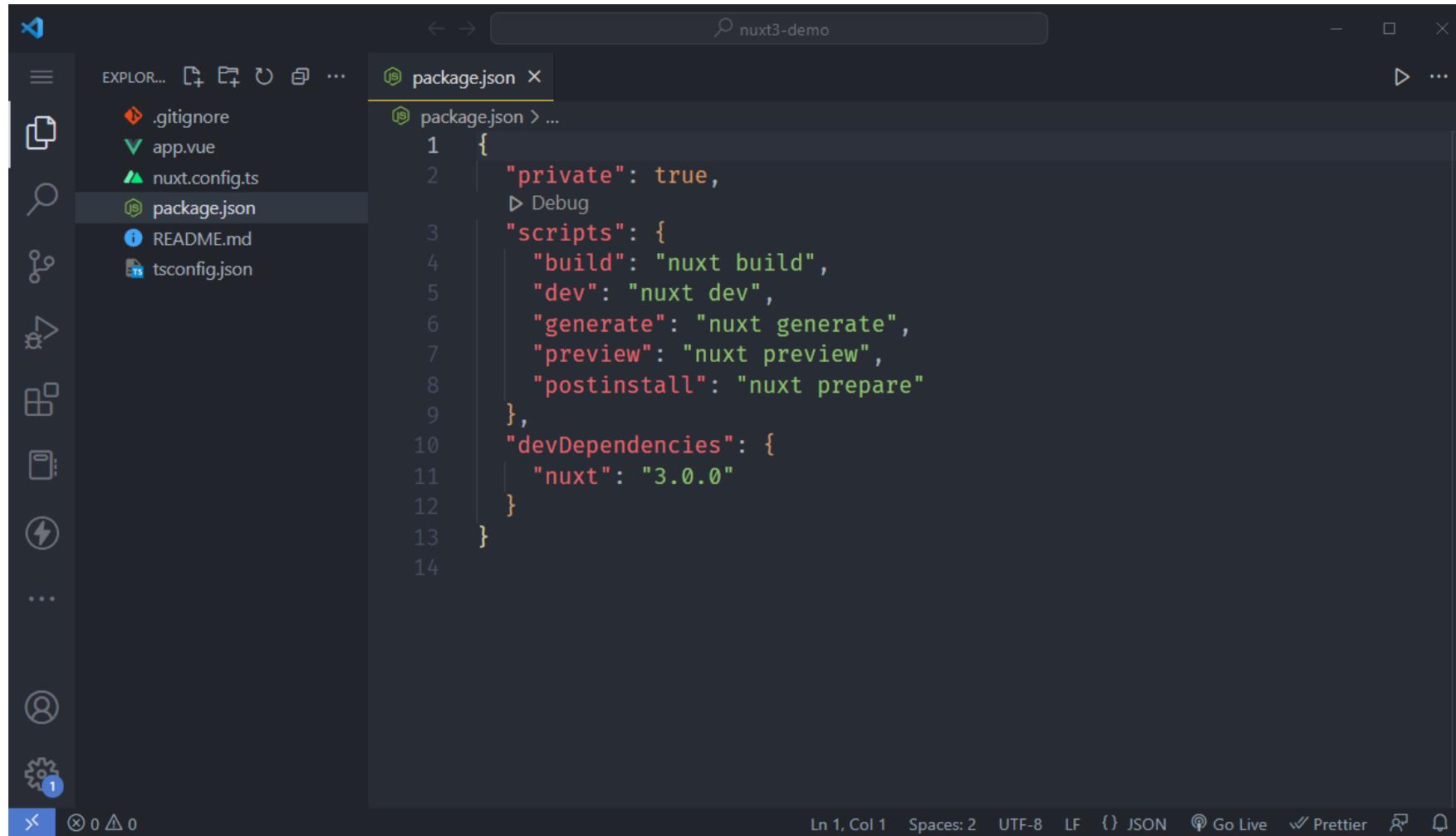
```
$ npm install
```

ติดตั้ง dependencies

```
$ npm run dev
```

คำสั่ง run โปรเจกต์ในโหมด dev

Nuxt 3 Starter Template Project



The screenshot shows a dark-themed instance of Visual Studio Code (VS Code) with the title bar "nuxt3-demo". The left sidebar contains icons for Explorer, Search, Problems, and other tools. The main area displays the contents of the "package.json" file, which is currently selected in the Explorer sidebar. The code is as follows:

```
1  {
2    "private": true,
3    "scripts": {
4      "build": "nuxt build",
5      "dev": "nuxt dev",
6      "generate": "nuxt generate",
7      "preview": "nuxt preview",
8      "postinstall": "nuxt prepare"
9    },
10   "devDependencies": {
11     "nuxt": "3.0.0"
12   }
13 }
14 }
```

The status bar at the bottom shows "Ln 1, Col 1" and "Spaces: 2". Other status indicators include "UTF-8", "LF", "JSON", "Go Live", "Prettier", and notifications.

Nuxt 3 Starter Template Project

The screenshot shows a dark-themed instance of Visual Studio Code. The left sidebar contains icons for file operations like Open, Save, Find, and Refresh. The Explorer sidebar on the left shows the project structure, including a .nuxt folder with types, app.config.mjs, components.d.ts, imports.d.ts, nuxt.d.ts, and tsconfig.json; a node_modules folder containing .gitignore, app.vue, nuxt.config.ts, package-lock.json, package.json, README.md, and tsconfig.json; and a .gitignore file. The main editor area displays the package.json file with the following content:

```
1 {  
2   "private": true,  
3   "scripts": {  
4     "build": "nuxt build",  
5     "dev": "nuxt dev",  
6     "generate": "nuxt generate",  
7     "preview": "nuxt preview",  
8     "postinstall": "nuxt prepare"  
9   },  
10  "devDependencies": {  
11    "nuxt": "3.0.0"  
12  }  
13}  
14
```

Below the editor, the terminal tab is active, showing the output of an npm install command:

```
added 608 packages, and audited 609 packages in 59s  
91 packages are looking for funding  
  run `npm fund` for details  
found 0 vulnerabilities
```

\$ npm install

ติดตั้ง dependencies

Nuxt 3 Run Project

The screenshot shows the VS Code interface with the following details:

- File Explorer:** Shows the project structure with files like .nuxt, app.config.mjs, components.d.ts, imports.d.ts, nuxt.d.ts, nux.json, tsconfig.json, node_modules, .gitignore, app.vue, nuxt.config.ts, package-lock.json, package.json, README.md, and tsconfig.json.
- Code Editor:** Displays the package.json file content:

```
3 "scripts": {  
4   "build": "nuxt build",  
5   "dev": "nuxt dev",  
6   "generate": "nuxt generate",  
7   "preview": "nuxt preview",  
8   "postinstall": "nuxt prepare"  
9 },  
10 "devDependencies": {  
11   "nuxt": "3.0.0"  
12 }  
13 }
```

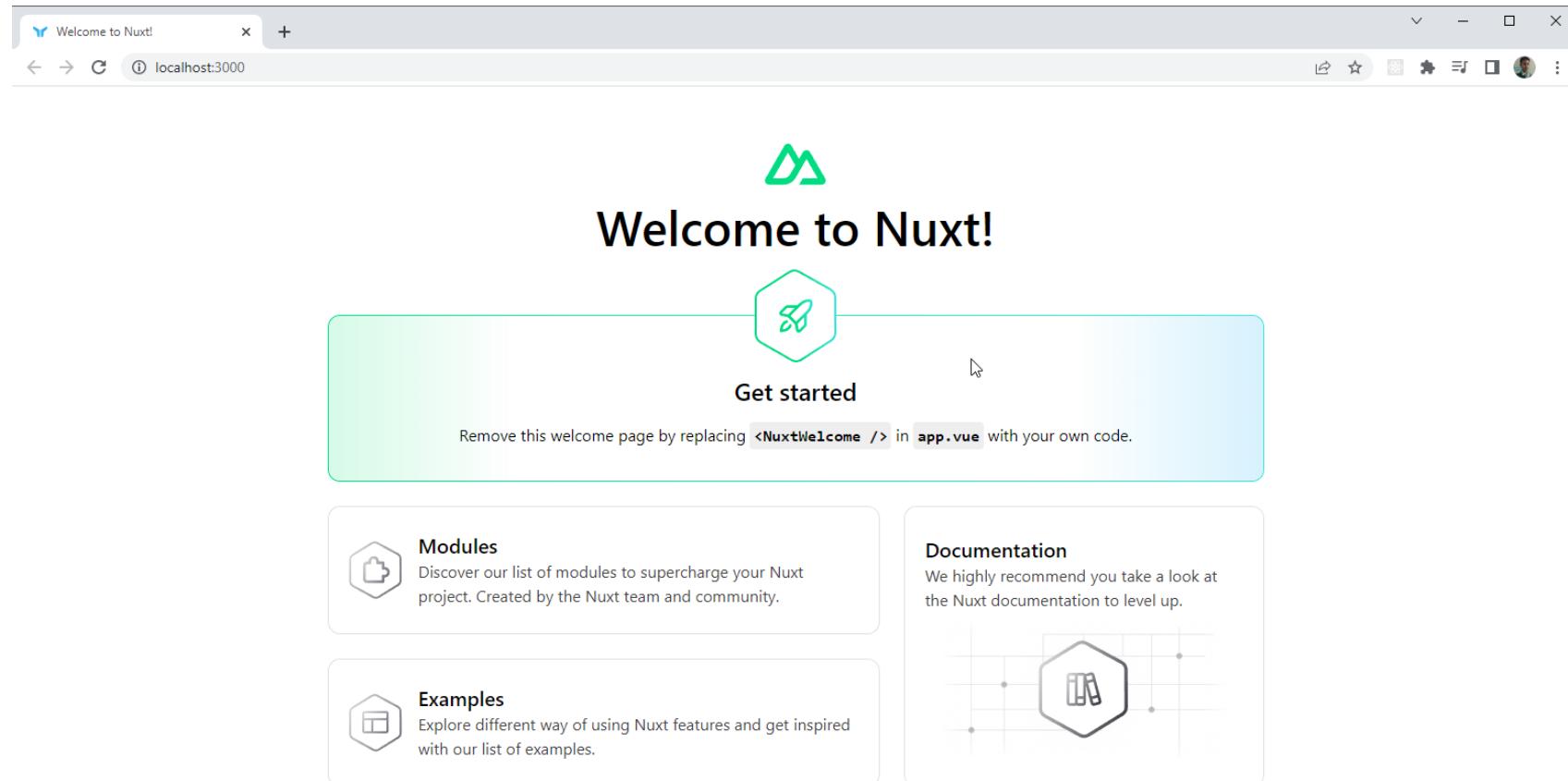
- Terminal:** Shows the output of the npm run dev command:

```
> Local: http://localhost:3000/  
> Network: http://172.18.176.1:3000/  
> Network: http://172.31.240.1:3000/  
> Network: http://192.168.1.146:3000/  
> Network: http://[2001:fb1:b5:3b3:19f:570:e1fb:394e]:3000/  
> Network: http://[2001:fb1:b5:3b3:e8e1:f2c3:5887:6291]:3000/  
  
i Vite client warmed up in 1409ms  
✓ Nitro built in 530 ms
```

\$ npm run dev

คำสั่ง run โปรเจ็คต์ในโหมด dev

Nuxt 3 Run Project



\$ npm run dev

คำสั่ง run โปรเจ็คต์ในโหมด dev

Hello Nuxt 3

The screenshot shows a development setup for a Nuxt 3 application. On the left is a dark-themed VS Code interface. The Explorer sidebar lists files like .nuxt, node_modules, .gitignore, app.vue, nuxt.config.ts, package-lock.json, package.json, README.md, and tsconfig.json. The app.vue component is open in the editor, displaying the following Vue template:

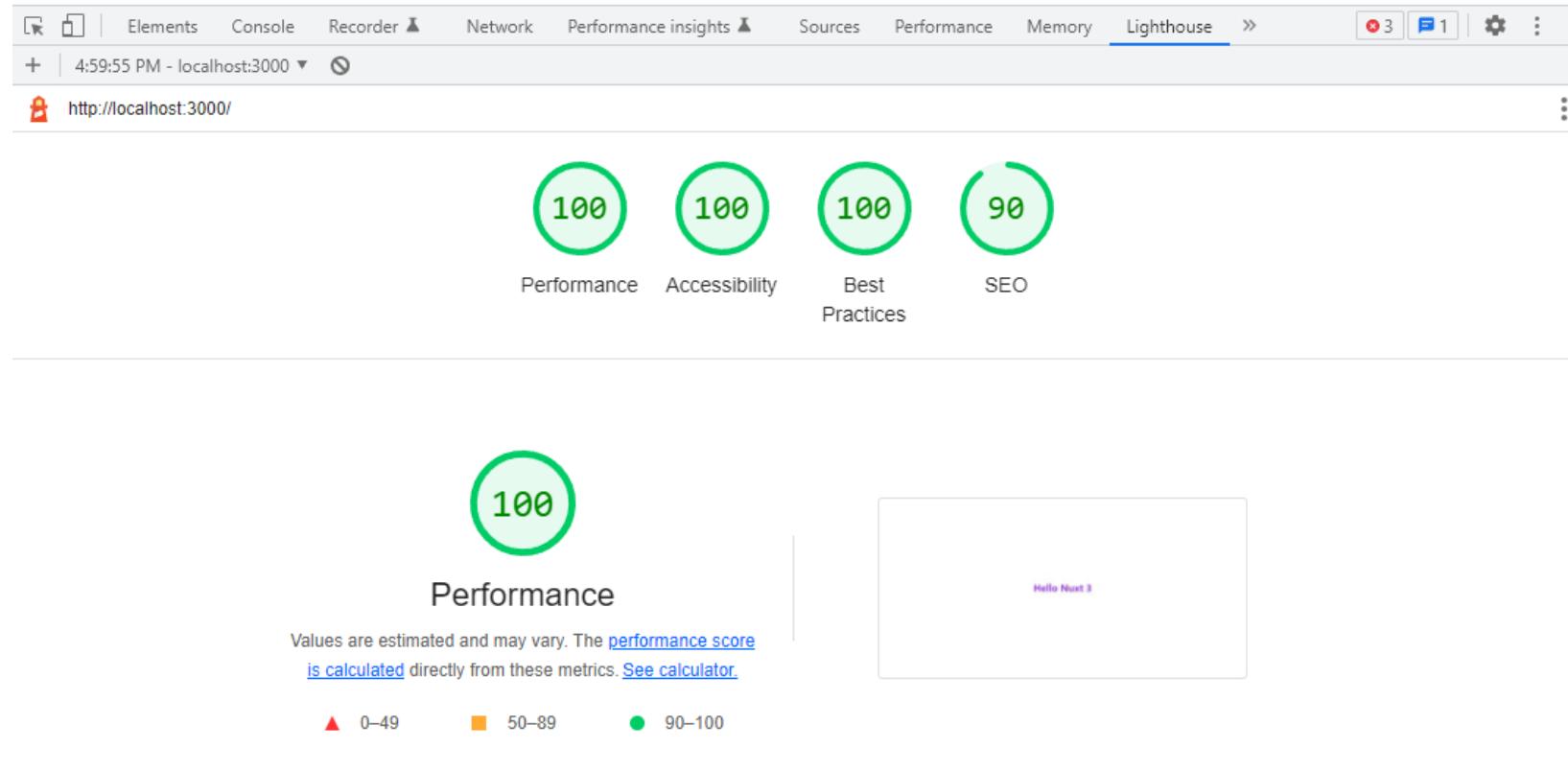
```
<template>
  <div>
    <!-- <NuxtWelcome /> -->
    <h1>Hello Nuxt 3</h1>
  </div>
</template>
```

The terminal at the bottom shows the build and start process:

```
> Local: http://localhost:3000/
> Network: http://172.18.176.1:3000/
> Network: http://172.31.240.1:3000/
> Network: http://192.168.1.146:3000/
> Network: http://[2001:fb1:b5:3b3:19f:570:e1fb:394e]:3000/
> Network: http://[2001:fb1:b5:3b3:e8e1:f2c3:5887:6291]:3000/

i Vite client warmed up in 1409ms
Nitro built in 530 ms
```

To the right is a browser window titled "Welcome to Nuxt!" showing the page "localhost:3000" with the text "Hello Nuxt 3".



Section 2

การจัดการข้อผิดพลาดฝั่ง Client และฝั่ง Server

- การเพิ่มหน้า Page not found (404)
- การใช้ useError Composable
- การทำ Dynamic Rendering Card View
- แสดงข้อผิดพลาดจากฝั่ง Server (Throwing a custom server error)
- แสดงข้อผิดพลาดจากฝั่ง Client (Throwing a Client side error)
- การ Handle Error ด้วย NuxtErrorBoundary



LIVE

จัดอบรมออนไลน์

Advanced Nuxt 3

ขั้นสูง

🗓 21,28 Jan , 4,11,18 Feb 2023
⌚ 20.00-23.30 PM

5 วัน
18 ชั่วโมงเต็ม



สอนสดผ่าน Zoom
รับจำนวนจำกัด



วันที่ 2



มีวิดีโอบันทึกอบรม
ย้อนหลังให้ทุกวัน

Section 3

การปรับปรุงฟังก์ชันด้วยการใช้ Utilities

- การปรับประสิทธิภาพของรูป (optimizing images) ด้วย NuxtImg
- การใช้ useState Composable
- การใช้งาน Vueuse Module
- การ Forcing Client Side Rendering



Section 4

การเขียนเว็บและหลายภาษาด้วย Nuxt 3 i18n

- พื้นฐาน Localization
- การติดตั้ง library @nuxtjs/i18n
- แนวทางการแปลภาษาใน Vue components
- External JSON or JS files
- Named formatting
- Pluralization
- Number localization
- Date and time localization
- Nuxt-specific localization
- Search engine optimization (SEO)



LIVE

จัดอบรมออนไลน์

Advanced Nuxt 3

ขั้นสูง

21,28 Jan , 4,11,18 Feb 2023

20.00-23.30 PM



สอนสดผ่าน Zoom
รับจำนวนจำกัด



5 วัน
18 ชั่วโมงเต็ม

วันที่ 3



มีวิดีโอบันทึกอบรม
ย้อนหลังให้ทุกวัน

Section 5

การสร้าง API ด้วย Nuxt 3

- Create a new nuxt 3 project
- Creating server routes
- Catching all routes
- Creating Middleware and Utilities
- Testing API



Section 6

Nuxt 3 Authentication and Middleware

- แนะนำ Supabase เป็นต้น
- การสมัครใช้งาน Supabase
- การเชื่อมต่อโปรเจกต์ Nuxt กับ Supabase
- เปิดใช้งาน Google Authentication
- Implementing Google OAuth
- Conditionally Rendering UI Based on Auth Status
- Implementing the Logout Functionality
- Protecting Routes With Inline Route Middleware
- Named Middleware
- Global Middleware



Section 7

Nuxt 3 JWT authentication

- Install the required packages
- Configure @nuxtjs/auth module in nuxt.config.js file
- Create login form dispatch nuxtClientInit and pass JWT token
- Use \$auth check authenticated and access the user's data
- Use auth middleware to protect routes
- Handle logout method on the \$auth object



LIVE

จัดอบรมออนไลน์

Advanced Nuxt 3

ขั้นสูง

21,28 Jan , 4,11,18 Feb 2023

20.00-23.30 PM



สอนสดผ่าน Zoom
รับจำนวนจำกัด



5 วัน
18 ชั่วโมงเต็ม

วันที่

4

มีวิดีโอบันทึกอบรม
ย้อนหลังให้ทุกวัน

Section 8

Nuxt 3 with Data Fetching and HTTP Requests

- ແນະໜ້າ Server Endpoints API
- Extracting the Path Parameters
- Extracting the Query Parameters
- Fetching the Data From the Client
- Fetch Composable
- Fixing an Issue With Error Handling



Section 9

Nuxt 3 with Prisma in Postgres database

- ແບະນຳໃຫ້ຮູ້ຈັກກັບ Prisma ORM
- Installing an ORM with Prisma
- Programmatically Creating Our Tables
- HTTP Verbs
- Fetching Data From the Database
- Deleting Data From the Database
- Adding the Other Two Endpoints
- Complex Filters
- Making a POST Request From the Client
- Making a GET Request From the Client
- Making a DELETE Request From the Client



LIVE

จัดอบรมออนไลน์

Advanced Nuxt 3

ขั้นสูง

21,28 Jan , 4,11,18 Feb 2023

20.00-23.30 PM



สอนสดผ่าน Zoom
รับจำนวนจำกัด



5 วัน
18 ชั่วโมงเต็ม

วันที่ 5



มีวิดีโอบันทึกอบรม
ย้อนหลังให้ทุกวัน

Section 10

Nuxt 3 State management with Pinia

- Installing Pinia in Nuxt 3
- Add Pinia to your nuxt.config file
- Build your Pinia store
- Bring Pinia in Vue Component



Section 11

การ Build และ Deployed project ไปใช้งานจริง

- Deploy Nuxt 3 app to Vercel
- Deploy Nuxt 3 app to Netlify
- Deploy Nuxt 3 app to Render
- Deploy Nuxt 3 sites with universal rendering on Azure





อาจารย์سامิต โกยม

สถาบันไอทีจีเนียส



samitkoyom@gmail.com



iamsamit



faeacebook.com/iamsamit



twitter.com/iamsamit



github.com/iamsamitdev



linkedin.com/in/samit-koyom-72173348