

สร้างแอปพลิเคชันแบบมือโปร ด้วย

Next.js & ChatGPT

- การใช้ Next.js สร้างแอปพลิเคชันทั้งในฝั่ง Frontend และ Backend
- เทคนิคการนำ ChatGPT และ ChatGPT Plus มาช่วยพัฒนาโปรแกรม
- การใช้ Next.js ร่วมกับฐานข้อมูล MongoDB และ Firebase Firestore Database
- สร้างระบบยืนยันตัวตนโดยใช้ Next-Auth และ Firebase Authentication
- การใช้ Tailwind CSS ปรับแต่งแอปพลิเคชันในแบบ Responsive Design
- อธิบายครบทุกเรื่องเกี่ยวกับ Next.js เช่น Server Actions, Middleware และอื่น ๆ

จิราวุธ วารินทร์

SCAN FREE



ไฟล์ประกอบหนังสือ

คำนำ >

Next.js เป็นเฟรมเวิร์คที่มีพื้นฐานมาจาก React โดยได้เพิ่มคุณสมบัติต่าง ๆ มากมาย ที่ช่วยให้ นักพัฒนาสามารถสร้างแอปพลิเคชันในโลกแห่งความจริงได้ง่ายยิ่งขึ้น เช่น สามารถสร้างเว็บแอปพลิเคชันได้ทั้งฝั่ง Frontend และ Backend สามารถเรนเดอร์เนื้อหาบางส่วนเตรียมไว้ก่อน (Pre-rendering) มีระบบจัดเก็บข้อมูลลงในแคช ทำให้หน้าเว็บเพจสามารถแสดงผลได้สมบูรณ์ตั้งแต่การโหลดใช้งานในครั้งแรก (คุณสมบัติ Server-Side Rendering และ Static Site Generation) ฯลฯ

เพื่อให้ผู้อ่านสามารถเข้าใจ Next.js ได้ครบทุกแง่มุม หนังสือเล่มนี้จึงถูกแบ่งออกเป็น 3 ส่วน ดังนี้

ส่วนที่ 1 เป็นการปูพื้นฐาน React ที่จำเป็นต้องใช้งานกับ Next.js เช่น การสร้างคอมโพเนนต์, การส่งผ่านข้อมูลโดยใช้ props, การใช้งานฮุก (Hook), การแชร์ข้อมูลผ่าน Context API และอื่น ๆ อีกมากมาย หลังจากอ่านส่วนนี้จบ ผู้อ่านจะมีความรู้พื้นฐานเกี่ยวกับ React ที่จำเป็นสำหรับการใช้งานร่วมกับ Next.js ได้แล้ว

ส่วนที่ 2 ได้แนะนำคุณสมบัติสำคัญของ Next.js ซึ่งช่วยให้นักพัฒนาสามารถสร้างเว็บแอปพลิเคชันทั้งในฝั่งไคลเอนต์และเซิร์ฟเวอร์ได้ เช่น การกำหนดเราท์ติ้ง, การ fetch ข้อมูลที่สามารถเลือกเก็บข้อมูลลงแคช, การใช้งาน Server Actions, การอ่านเขียนข้อมูลไปยังฐานข้อมูล MongoDB, การใช้งาน Middleware, การยืนยันตัวตนโดยใช้ Next-Auth ฯลฯ

ส่วนที่ 3 เป็นการนำ ChatGPT มาช่วยสร้างแอปพลิเคชันร่วมกับ Next.js โดยจะเริ่มจากแนะนำความสามารถของ ChatGPT ตามด้วยตัวอย่างการนำ ChatGPT มาช่วยในการเขียนโค้ดและแก้ไขโค้ด ในส่วนนี้ได้ยกตัวอย่างการนำ ChatGPT มาใช้สร้างแอปพลิเคชันสำหรับใช้บริการต่าง ๆ จาก Firebase อาทิเช่น การอ่านเขียนข้อมูลในแบบเรียลไทม์กับ Firebase Cloud Firestore, การฝากไฟล์ไว้กับ Firebase Storage และวิธีสร้างระบบยืนยันตัวตนโดยใช้ Firebase Authentication

จากเนื้อหาทั้งหมดของหนังสือเล่มนี้ ที่ครอบคลุมส่วนที่จำเป็นทั้งหมดของ Next.js พร้อมกับแสดงตัวอย่างโค้ดจำนวนมากที่มีคำอธิบายโดยละเอียด ผู้เขียนจึงมั่นใจเป็นอย่างยิ่งว่า ผู้อ่านจะมีความรู้เกี่ยวกับ Next.js เพียงพอสำหรับการนำไปสร้างแอปพลิเคชันจริงได้อย่างแน่นอน

จิราวุธ วารินทร์

jeerawuth@me.com

สารบัญ

Part 1 Basic React for Next.js		
บทที่ 1	แนะนำ Next.js	6
บทที่ 2	คอมโพเนนต์และหน้าเว็บเพจ	25
บทที่ 3	การกำหนดสไตล์และการใช้งาน Tailwind CSS	42
บทที่ 4	การใช้งาน props	72
บทที่ 5	พื้นฐานการใช้งาน state	88
บทที่ 6	การใช้ฮุคในฟังก์ชันคอมโพเนนต์	111
บทที่ 7	พื้นฐานการใช้งาน Context API	141
Part 2 Essential Next.js		
บทที่ 8	การใช้งานเร้าท์ติ้งใน Next.js	166
บทที่ 9	การกำหนด Layout และเมนูใน Next.js	195
บทที่ 10	วิธี fetch ข้อมูลมาใช้ในคอมโพเนนต์	231
บทที่ 11	แบบฟอร์มและการใช้งาน Server Actions	260
บทที่ 12	Metadata และ Image	288
บทที่ 13	Next.js กับฐานข้อมูล MongoDB	300
บทที่ 14	Route Handler ใน Next.js	347
บทที่ 15	ตัวอย่างการสร้างและใช้งาน API	376
บทที่ 16	การใช้งาน Middleware ใน Next.js	409
บทที่ 17	ยืนยันตัวตนโดยใช้ Next-Auth	426
Part 3 Create Project with ChatGPT		
บทที่ 18	ChatGPT กับการพัฒนาโปรแกรม	454
บทที่ 19	เรียนรู้ Firebase ด้วย ChatGPT	485
บทที่ 20	การสั่ง Build และ Deploy แอปพลิเคชัน	534

Part 1

Basic React
for Next.js

01 | แนะนำ Next.js

Next.js คือ เฟรมเวิร์คที่ช่วยให้เราสามารถสร้างแอปพลิเคชันด้วย React ทำได้ง่ายและรวดเร็วยิ่งขึ้น โดย Next.js ถูกสร้างมาเพื่อรองรับการใช้งานจริง (Productions) โดยเฉพาะแอปพลิเคชันที่มีผู้ใช้งานจำนวนมาก

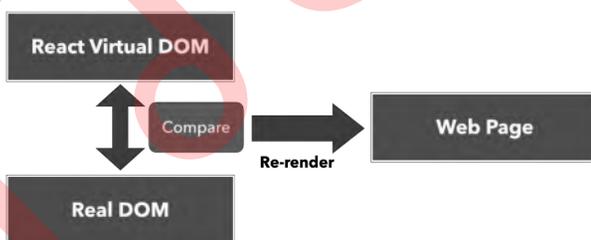
Next.js ได้เตรียมคุณสมบัติหลายอย่างซึ่งช่วยให้การโหลดเว็บเพจทำได้เร็ว โดยสามารถเรนเดอร์เนื้อหาบางส่วนเตรียมไว้ก่อน (Pre-rendering) และมีระบบการจัดเก็บข้อมูลลงในแคช ทำให้หน้าเว็บเพจสามารถแสดงผลได้สมบูรณ์ตั้งแต่การโหลดใช้งานในครั้งแรก (คุณสมบัติ Server-Side Rendering และ Static Site Generation) ด้วยการที่เว็บเพจมีเนื้อหาตั้งแต่การโหลดครั้งแรก จึงทำให้เว็บเพจสามารถติดอันดับการค้นหาใน Google หรือ Search Engine อื่น ๆ ได้ง่ายขึ้นด้วย

บทวนเกี่ยวกับ React

React คือ JavaScript ไกลบรารีที่ถูกพัฒนาขึ้นโดย Facebook มีจุดประสงค์สำหรับสร้างส่วนติดต่อกับผู้ใช้ (User Interface หรือ UI) บนเว็บเพจหรือบนเว็บแอปพลิเคชัน

React จะแบ่งเว็บเพจออกเป็นส่วนย่อย เรียกว่า คอมโพเนนต์ (Components) นักพัฒนาสามารถนำคอมโพเนนต์ต่าง ๆ มาประกอบเป็นหน้าเว็บเพจ หากหน้าเพจมีเนื้อหาซ้ำก็สามารถนำคอมโพเนนต์เก่ามาใช้งานได้ ด้วยการแบ่งเนื้อหาออกเป็นส่วนย่อย จึงทำให้การพัฒนาแอปพลิเคชันทำได้ง่ายกว่าการเขียนโค้ด JavaScript ด้วยตนเองทั้งหมด

React จะใช้วิธีเก็บโครงสร้างของเว็บเพจไว้ในหน่วยความจำ เรียกว่า Virtual DOM โดยจะนำไปเปรียบเทียบกับเอลิเมนต์ที่อยู่ใน DOM จริง ด้วยวิธีนี้ React จึงสามารถอัปเดตเฉพาะส่วนของเว็บเพจที่มีการเปลี่ยนแปลงไปจากเดิม โดยไม่จำเป็นต้องอัปเดตข้อมูลทั้งหน้าเว็บเพจ ทำให้การแสดงผลและการโต้ตอบระหว่างผู้ใช้ในระหว่างใช้งานแอปพลิเคชันสามารถทำได้อย่างรวดเร็วยิ่งขึ้น



ด้วยคุณสมบัติต่าง ๆ ของ React นักพัฒนาจึงนิยมนำ React มาใช้สร้างเว็บแอปพลิเคชันที่ใช้เพียงหน้าเว็บเพจเดียว (เรียกว่า Single Page Application : SPA) ที่สามารถแสดงผลและโต้ตอบกับผู้ใช้โดยไม่ต้องโหลดหน้าเว็บใหม่ ทำให้แอปพลิเคชันทำงานได้รวดเร็วและมีประสิทธิภาพ

จาก React มาสู่ Next.js

React เป็น JavaScript โลบรารี ที่ใช้งานง่ายและมีประสิทธิภาพสูง จึงเป็นที่นิยมและถูกนำไปใช้งานอย่างกว้างขวาง แต่การนำ React ไปใช้งานจริงยังมีข้อจำกัดบางประการ เช่น ไม่มีกำหนดแนวทางหรือโครงสร้างใด ๆ มาให้ นักพัฒนาจะต้องออกแบบเอง ดังนั้น หากนักพัฒนาออกแบบโครงสร้างของแอปพลิเคชันไม่ดีพอตั้งแต่แรก ก็จะทำให้การแก้ไขหรือการขยายส่วนประกอบต่าง ๆ ของแอปพลิเคชันทำได้ยาก

React ใช้วิธีเรนเดอร์เนื้อหาในฝั่งไคลเอนต์เป็นหลัก ทำให้เว็บเพจไม่มีข้อมูลที่ใช้อธิบายว่าเว็บเพจปัจจุบันมีเนื้อหาเกี่ยวกับอะไร จึงมีปัญหากับการทำ SEO ยกต่อการทำให้เว็บไซต์ติดอันดับการค้นหาใน Google หรือใน Search Engine ต่าง ๆ

เพื่อแก้ปัญหานี้ React เฟรมเวิร์คที่ชื่อว่า Next.js จึงถูกสร้างขึ้น โดยนำ React มาปรับปรุงใหม่ เพื่อให้สามารถสร้างแอปพลิเคชันสำหรับการใช้งานในโลกแห่งความจริงได้ดียิ่งขึ้น

Next.js ถูกพัฒนาขึ้นโดย Vercel และถูกเผยแพร่ครั้งแรกในปี ค.ศ. 2017 ซึ่ง Next.js ยังคงใช้รูปแบบพื้นฐานของ React เอาไว้ เช่น ใช้ JSX สำหรับสร้างเนื้อหาของคอมโพเนนต์, ส่งผ่านข้อมูลระหว่างคอมโพเนนต์โดยใช้ props, การจัดการ state และการใช้ฮุค ฯลฯ เรียกได้ว่านักพัฒนาที่คุ้นเคยกับ React มาก่อนแล้ว ก็สามารถเรียนรู้การใช้งาน Next.js ได้อย่างรวดเร็ว

จากที่กล่าวมาสามารถสรุปได้ว่า Next.js ก็คือ เครื่องมือสำหรับการพัฒนาเว็บแอปพลิเคชันด้วย React ในแบบมืออาชีพ ที่สามารถนำแอปพลิเคชันไปใช้งานจริงได้นั่นเอง

คุณสมบัติพื้นฐานของ Next.js

Next.js คือ JavaScript เฟรมเวิร์ค ซึ่งพัฒนาต่อยอดมาจาก React โดยมีคุณสมบัติหลัก ๆ ดังนี้

- **TypeScript Support** คือ รองรับการเขียนโค้ดด้วย TypeScript โดยไม่ต้องตั้งค่าส่วนประกอบใด ๆ
- **Server-Side Rendering (SSR)** คือ การสร้างเนื้อหาเว็บเพจในฝั่งเซิร์ฟเวอร์ จากนั้นจึงส่ง HTML ที่สมบูรณ์แล้วไปแสดงผลยังไคลเอนต์ ด้วยวิธีนี้การโหลดหน้าเว็บเพจในครั้งแรกจึงทำได้เร็ว และยังช่วยในเรื่องของการทำ SEO เพิ่มอันดับการค้นหาใน Google หรือ Search Engine อื่น ๆ
- **Static Site Generation (SSG)** คือ การสร้างหน้าเว็บเตรียมเอาไว้ล่วงหน้าตั้งแต่ตอนสร้างแอปพลิเคชัน (Build Time) เมื่อผู้ใช้งานต้องการดูหน้าเว็บเพจ เซิร์ฟเวอร์สามารถส่งผลลัพธ์ไปแสดงผลได้ทันทีโดยไม่ต้องรอประมวลผล
- **Code Splitting** คือ การแบ่งโค้ดออกเป็นส่วนเล็ก ๆ (เรียกว่า chunks) เพื่อลดขนาดของไฟล์ ทำให้เว็บไซต์โหลดเร็วขึ้นและประหยัดทรัพยากร
- **Image Optimization** คือ ระบบจัดการกับรูปภาพ ที่สามารถปรับรูปภาพให้มีขนาดและความละเอียดที่เหมาะสมตามสภาพแวดล้อมในขณะนั้น ๆ

- File-base Routing คือ การกำหนดระบบเราท์ติ้งโดยใช้ชื่อโพลเดอร์และโครงสร้างการจัดเรียงโพลเดอร์ ซึ่งวิธีนี้ช่วยให้การจัดโครงสร้างของแอปพลิเคชันสามารถทำได้ง่าย
- API Routes คือ รองรับการสร้าง API endpoints ได้ภายในโปรเจกต์ ทำให้สามารถสร้างเว็บแอปพลิเคชันทั้งในแบบ Frontend และ Backend โดยไม่ต้องติดตั้งส่วนประกอบเพิ่มเติม
- Hot Module Replacement (HMR) คือ คุณสมบัติในการแสดงผลลัพธ์หลังเปลี่ยนแปลงโค้ดในทันที โดยไม่ต้องรีเฟรชหน้าเว็บเพจ
- Built-In CSS and Sass Support คือ สนับสนุนการปรับแต่งสไตล์ด้วย CSS และ Sass รวมถึงสนับสนุนการใช้งาน CSS-in-JS ร่วมกับไลบรารีต่าง ๆ

เตรียมพร้อมก่อนใช้งาน Next.js

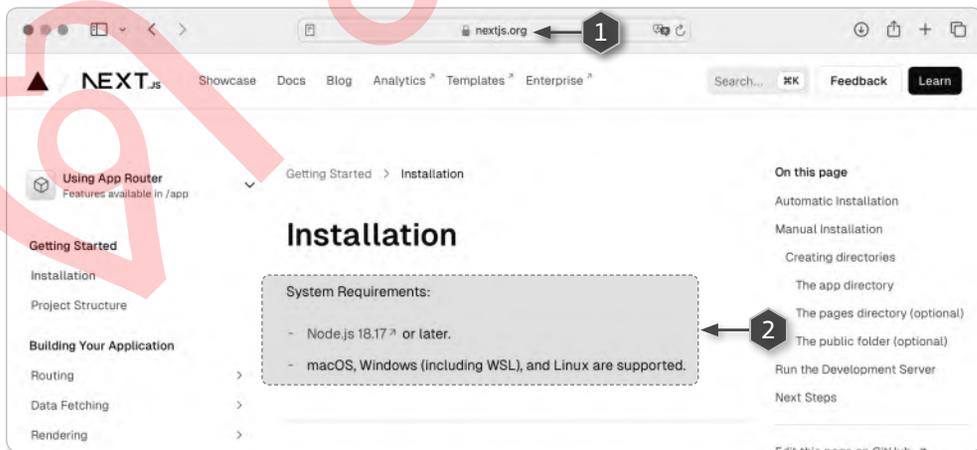
ก่อนใช้งาน Next.js แนะนำให้อ่านเอกสารเกี่ยวกับคุณสมบัติพื้นฐานของแต่ละเวอร์ชัน และติดตั้งซอฟต์แวร์หรือเครื่องมือบางอย่างลงในเครื่องคอมพิวเตอร์ ดังนี้

- ขั้นตอนที่ 1 อ่านเอกสารจาก nextjs.org
- ขั้นตอนที่ 2 ติดตั้ง Node.js
- ขั้นตอนที่ 3 ติดตั้ง IDE หรือ Code Editor สำหรับใช้แก้ไขโค้ดต่าง ๆ

ขั้นตอนที่ 1 อ่านเอกสารจาก nextjs.org

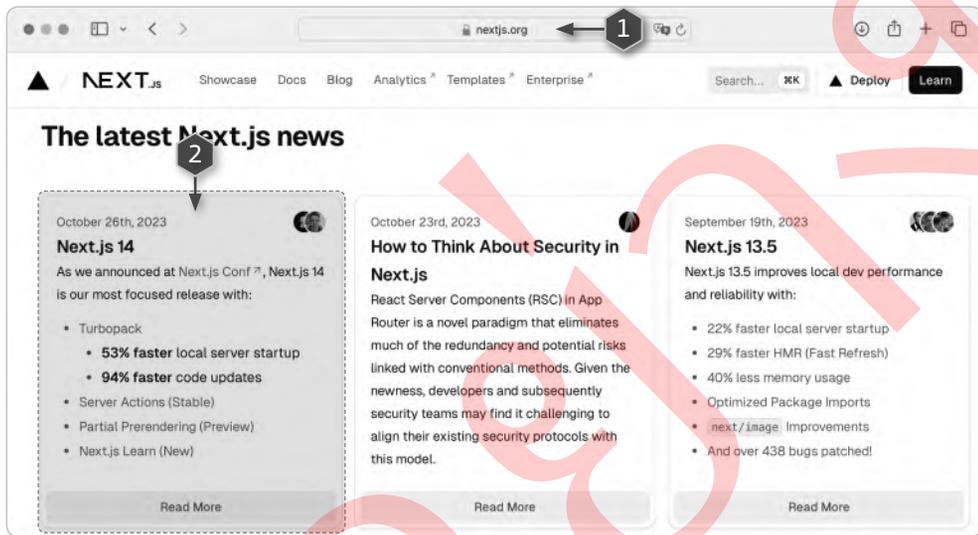
เนื่องจาก Next.js ถูกพัฒนาตลอดเวลา ดังนั้น ก่อนใช้งาน Next.js ทุกครั้งจะต้องตรวจสอบเวอร์ชัน และความต้องการของระบบที่ใช้กับเวอร์ชันนั้น ๆ เสียก่อน

1. ไปที่เว็บ <https://nextjs.org/docs/getting-started/installation>
2. ให้สังเกตในส่วนของ System Requirements ซึ่งบอกถึงความต้องการของระบบที่จะต้องใช้กับ Next.js มีอะไรบ้าง ซึ่งในตัวอย่าง การใช้ Next.js 14 จะต้องติดตั้ง Node.js เวอร์ชัน 18.17 ขึ้นไป โดยรองรับกับระบบปฏิบัติการแมคโอเอส, วินโดวส์ (ที่สนับสนุน WSL) และระบบปฏิบัติการลินุกซ์

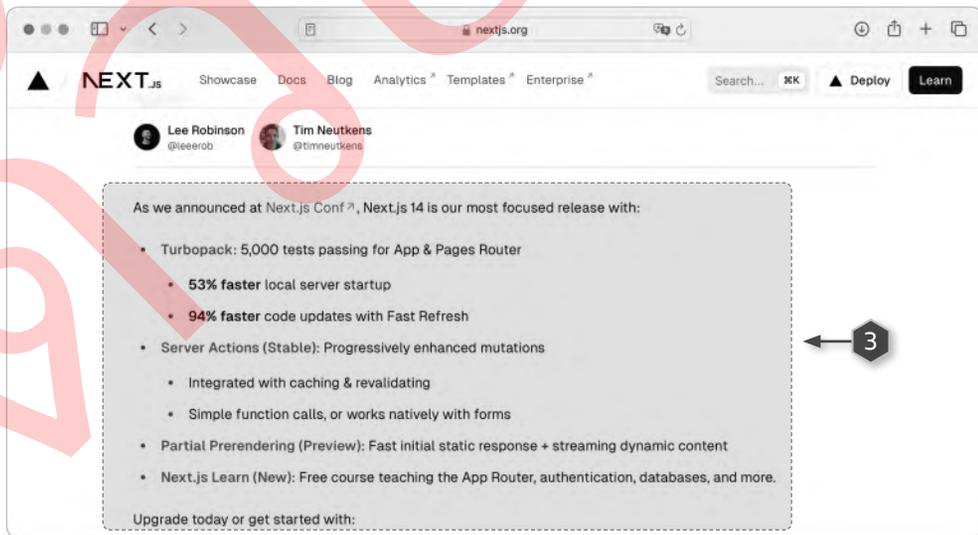


นอกจากต้องทราบความต้องการพื้นฐานแล้ว นักพัฒนาจะต้องทราบว่า Next.js เวอร์ชันที่ต้องการใช้งานนั้นมีคุณสมบัติอะไรบ้างที่ถูกประกาศใช้งานอย่างเป็นทางการ (stable) และคุณสมบัติใดบ้างที่อยู่ในขั้นตอนการพัฒนา (preview, alpha หรือ beta)

1. เปิดเบราว์เซอร์และไปยัง <https://nextjs.org/blog> เพื่อดูข่าวสารที่ทาง Next.js ประกาศออกมา
2. คลิกดูรายละเอียดในข้อมูล หรือเวอร์ชันที่สนใจ เช่น Next.js 14



3. จะแสดงรายละเอียดเกี่ยวกับเวอร์ชันแสดงให้เห็นทราบ เช่น ใน Next.js 14 สามารถใช้คุณสมบัติ Server Actions ได้ เนื่องจากผ่านการทดสอบว่าเสถียรแล้ว (Stable) ส่วนคุณสมบัติ Partial Prerendering ยังอยู่ในขั้นตอนการทดสอบในเบื้องต้น (Preview) จึงอาจพบปัญหาในระหว่างใช้งาน

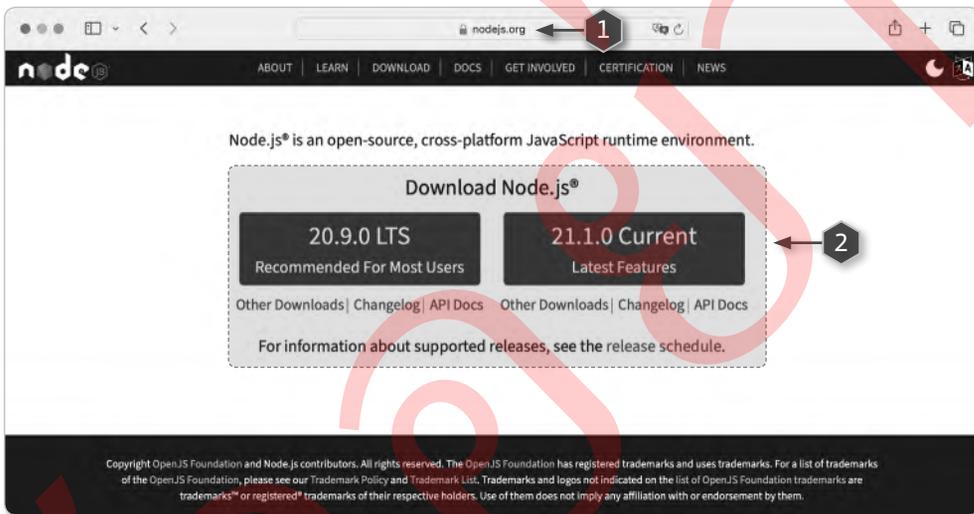


ขั้นตอนที่ 2 ติดตั้ง Node.js

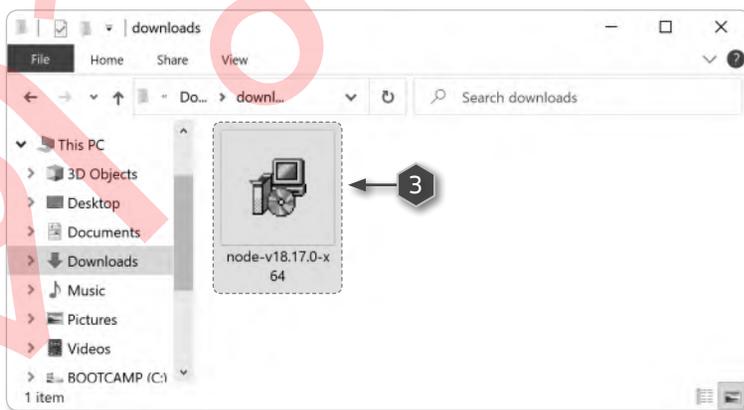
Node.js เป็นแพลตฟอร์มที่อนุญาตให้นักพัฒนาเขียนโค้ด JavaScript (หรือ TypeScript) สำหรับติดต่อกับระบบปฏิบัติการ หรือสร้างแอปพลิเคชันในฝั่งเซิร์ฟเวอร์ได้

เนื่องจาก Next.js ใช้คุณสมบัติพื้นฐานบางอย่างมาจาก Node.js ดังนั้น เราจึงต้องติดตั้ง Node.js ลงไปในเครื่องคอมพิวเตอร์เสียก่อน

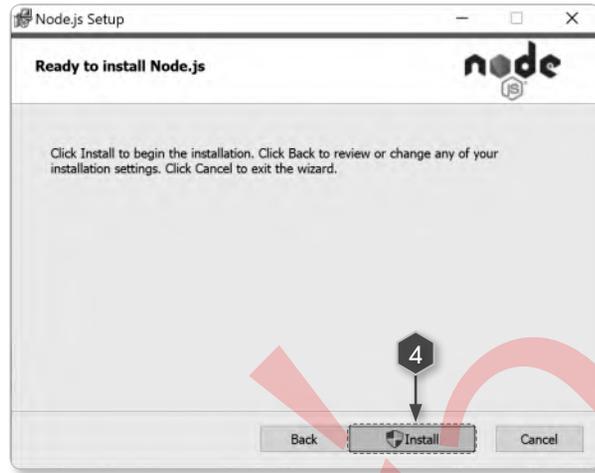
1. ไปที่เว็บไซต์ <https://nodejs.org/>
2. คลิกเพื่อดูดาวน์โหลด Node.js โดยมี 2 เวอร์ชันให้เลือก ได้แก่ เวอร์ชันที่เป็น LTS (Long Term Support) ซึ่งหมายถึงเวอร์ชันที่ถูกใช้งานอย่างแพร่หลายโดยไม่พบปัญหา กับเวอร์ชัน Current ที่หมายถึงเวอร์ชันล่าสุดในปัจจุบัน โดยส่วนตัวของผู้เขียนแนะนำให้ดาวน์โหลดเวอร์ชันที่เป็น LTS



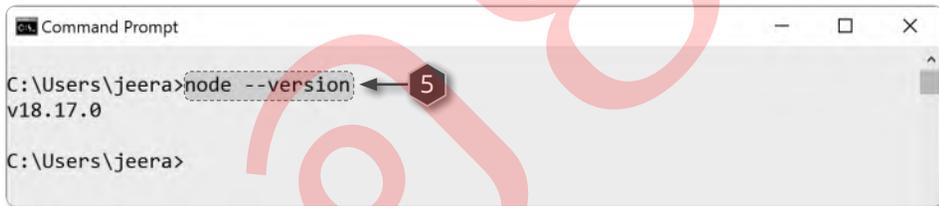
3. หลังจากดาวน์โหลดเรียบร้อยแล้ว ให้ดับเบิลคลิกเพื่อติดตั้ง Node.js



- ทำตามขั้นตอนที่ปรากฏบนหน้าจอจนกระทั่งติดตั้งเรียบร้อยแล้ว

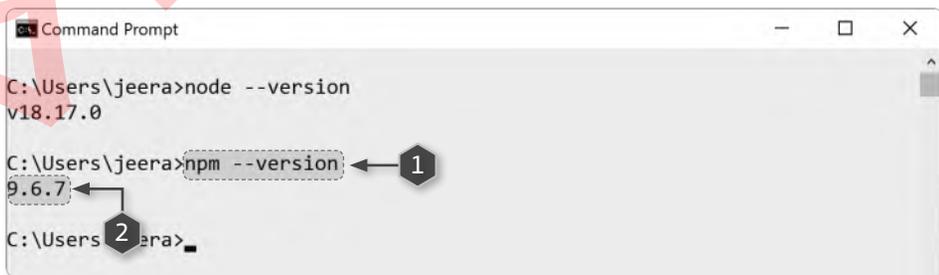


- เปิดหน้าต่าง Command Prompt ของวินโดวส์ หรือหน้าต่าง Terminal ในแมคโอเอส จากนั้นพิมพ์คำสั่ง `node --version` หรือ `node --v` เพื่อตรวจสอบเวอร์ชันของ Node.js ถ้าการติดตั้ง Node.js ถูกต้องสมบูรณ์จะพบหมายเลขเวอร์ชันแสดงให้ทราบ



หลังจากติดตั้ง Node.js เราจะได้เครื่องมือบางอย่างจาก Node.js มาใช้งาน เช่น ได้เครื่องมือ Node Package Manager หรือ npm สำหรับจัดการกับแพ็คเกจต่าง ๆ เช่น ใช้ติดตั้งแพ็คเกจ, ถอดแพ็คเกจ หรืออัปเดตแพ็คเกจ หากต้องการตรวจสอบว่า npm ได้ถูกติดตั้งเรียบร้อยแล้ว สามารถตรวจสอบจากหมายเลขเวอร์ชันของ npm ได้ ดังนี้

- เปิดหน้าต่าง Command Prompt ของวินโดวส์ หรือหน้าต่าง Terminal ในแมคโอเอส จากนั้นพิมพ์คำสั่ง `npm --version` หรือ `npm --v` เพื่อตรวจสอบเวอร์ชันของ npm
- ถ้าการติดตั้ง npm ถูกต้องสมบูรณ์จะพบหมายเลขเวอร์ชันของ npm แสดงให้ทราบ



NOTE\>

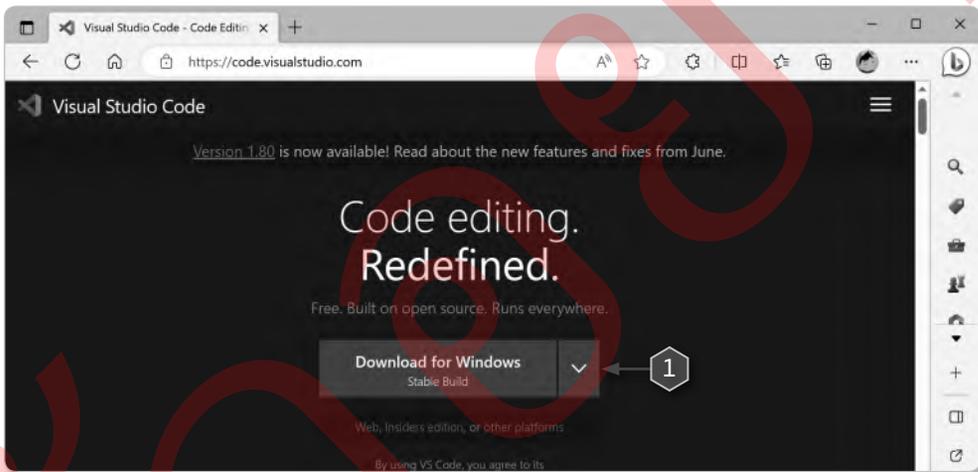
หากพบปัญหาเกี่ยวกับการใช้งาน npm ให้ติดตั้ง npm เข้าไปใหม่โดยใช้คำสั่ง `npm install npm -g`

ขั้นตอนที่ 3 ติดตั้ง Visual Studio Code

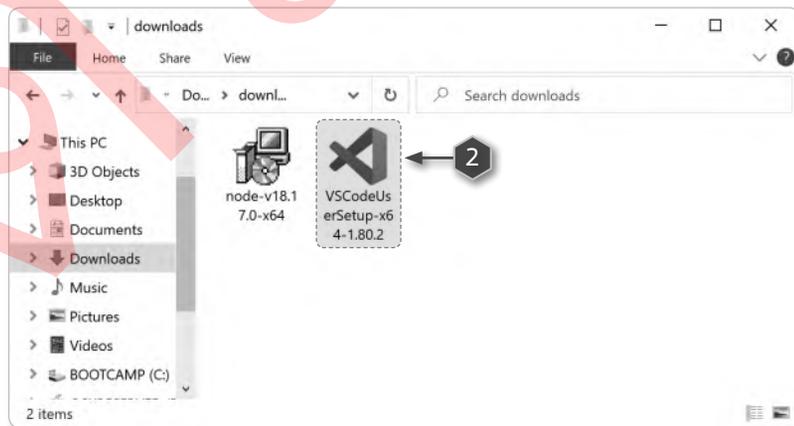
Visual Studio Code (หรือเรียกสั้น ๆ ว่า VS Code) เป็น Code Editor ยอดนิยมจากทางไมโครซอฟท์ ที่มีผู้ใช้งานเป็นจำนวนมากทั่วโลก โดยมีจุดเด่นมากมาย เช่น สามารถดาวน์โหลดไปใช้งานได้ฟรี, สามารถติดตั้งส่วนประกอบเสริมจึงเพิ่มความสามารถได้ไม่จำกัด, มีตัวช่วยรอกคำสั่ง, มีการใช้สีเพื่อแยกโค้ดคำสั่งแต่ละแบบจึงทำให้แก้ไขโค้ดหรืออ่านโค้ดได้ง่าย และที่สำคัญคือ สามารถใช้กับภาษาคอมพิวเตอร์ได้เกือบทุกภาษา ไม่ว่าจะเป็น C, Java, PHP, Python, JavaScript และภาษาอื่น ๆ อีกมากมาย

การติดตั้ง VS Code เพื่อใช้งานกับ JavaScript (หรือ TypeScript) มีขั้นตอน ดังนี้

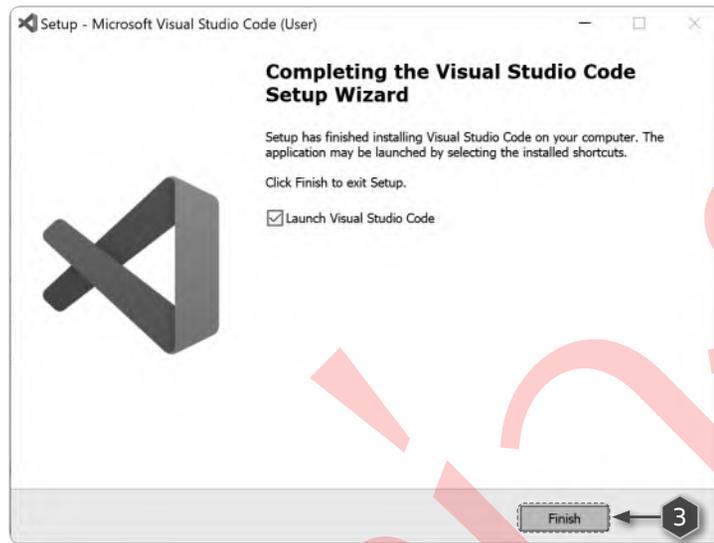
1. ไปที่เว็บไซต์ <https://code.visualstudio.com> คลิกปุ่ม Download



2. ดับเบิลคลิกไฟล์ติดตั้ง และทำตามขั้นตอนที่ปรากฏบนหน้าจอ



3. คลิกปุ่ม Finish เป็นขั้นตอนสุดท้าย

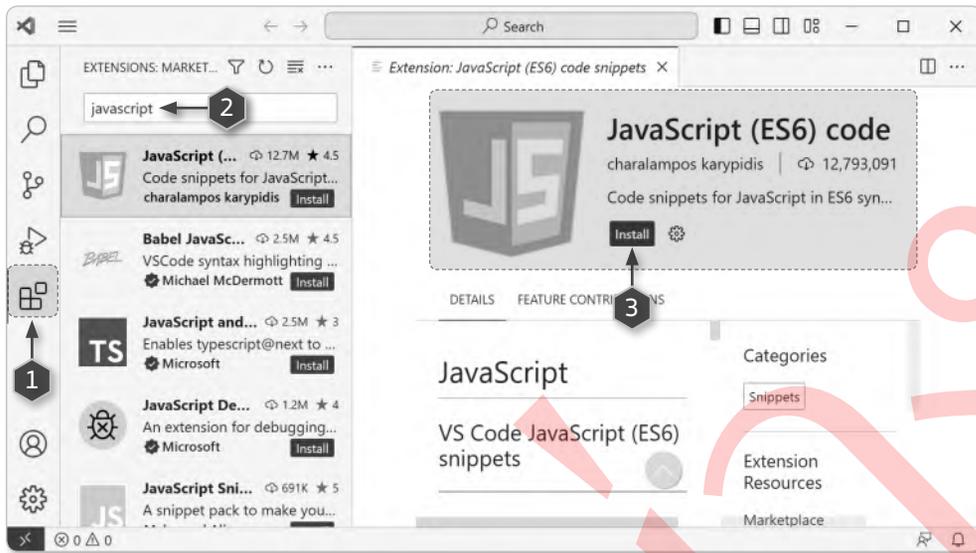


เพื่อความสะดวกในการพัฒนา Next.js ร่วมกับ Visual Studio Code แนะนำให้ติดตั้งส่วนประกอบเสริม (extensions หรือ plug-in) ต่อไปนี้ลงไป

- JavaScript (ES6) code snippets เป็นส่วนประกอบเสริมที่จะช่วยให้เราสามารถเขียนโค้ด JavaScript สมัยใหม่ (ES6 ขึ้นไป) ทำได้ง่ายและรวดเร็ว โดยในระหว่างที่กรอกโค้ดจะมีคำแนะนำเกี่ยวกับชุดคำสั่งหรือแสดงโครงสร้างโค้ดพื้นฐานมาให้อัตโนมัติ
- Prettier - Code formatter เป็นเครื่องมือสำหรับการจัดรูปแบบโค้ดโดยอัตโนมัติ เช่น มีการปรับการเยื้อง ทำให้โค้ดมีระเบียบและอ่านง่าย
- Tailwind CSS IntelliSense เป็นเครื่องมือที่ช่วยแนะนำชื่อคลาส (utilities class) ของ Tailwind CSS มาให้อัตโนมัติในระหว่างกรอกโค้ด พร้อมแสดงคำอธิบายโค้ด CSS และสามารถซ่อนชื่อคลาสอัตโนมัติ (หนังสือเล่มนี้ใช้วิธีปรับแต่งหน้าต่างของแอปพลิเคชันด้วย Tailwind CSS ดังนั้น จึงแนะนำให้ติดตั้งส่วนประกอบเสริมนี้ด้วย)

การติดตั้งส่วนประกอบเสริม (Extensions) เพิ่มเข้าไปใน VS Code มีขั้นตอน ดังนี้

1. เปิดโปรแกรม VS Code จากนั้นให้คลิกปุ่ม  หรือกดปุ่ม <Ctrl+Shift+X> (ในแมคให้กดปุ่ม <Cmd+Shift+X>) เพื่อแสดงส่วนประกอบเสริมทั้งหมดที่ได้ติดตั้งเอาไว้
2. ที่ช่องค้นหา ให้กรอก JavaScript เพื่อค้นหาส่วนประกอบเสริมสำหรับ JavaScript
3. จะพบส่วนประกอบเสริมมากมายที่เกี่ยวกับ JavaScript คลิกปุ่ม Install ที่ JavaScript (ES6) code snippets เพื่อติดตั้งส่วนประกอบต่าง ๆ ที่จำเป็นต่อการใช้ Visual Studio Code ร่วมกับ JavaScript



หลังจากติดตั้งส่วนประกอบเสริมไปแล้ว หากต้องการลบส่วนประกอบเสริมออก ก็ให้เลือกส่วนประกอบเสริมที่ต้องการลบ จากนั้นให้คลิกปุ่ม `uninstall`

1. แสดงรายการส่วนประกอบที่ได้ติดตั้งใน VS Code โดยคลิกปุ่ม  หรือกดปุ่ม `<Ctrl+Shift+X>` ในระบบปฏิบัติการวินโดวส์ (ในแมคให้กดปุ่ม `<Cmd+Shift+X>`)
2. คลิกเลือกส่วนประกอบเสริมที่ต้องการ
3. คลิกปุ่ม `uninstall` เพื่อถอดส่วนประกอบเสริมนั้นออกไป

สร้าง Next.js โดยรันแพ็คเกจ `create-next-app`

การสร้าง Next.js แอปพลิเคชัน สามารถทำได้โดยใช้เครื่องมือที่เรียกว่า `create-next-app` โดยรันคำสั่งต่อไปนี้

```
npx create-next-app@latest
```

- `npx` คือ คำสั่งที่รันแพ็คเกจ (เป็นคนละคำสั่งกับ `npm` ที่ใช้จัดการกับแพ็คเกจ)
- `create-next-app` คือ แพ็คเกจที่ใช้สร้าง Next.js แอปพลิเคชัน
- `@latest` คือ คำสั่งที่บอกให้ทราบว่าการใช้งาน Next.js เวอร์ชันล่าสุด

สร้างโปรเจกต์แรกกับ Next.js

ก่อนการสร้างโปรเจกต์ในครั้งแรก แนะนำให้ตรวจสอบว่าได้ติดตั้ง Node.js ไว้ในเครื่องเรียบร้อยแล้ว จากนั้นจึงสร้าง Next.js แอปพลิเคชันด้วยคำสั่ง `npx create-next-app@latest` หลังจากสร้างแอปพลิเคชันเสร็จก็สามารถทดลองรันแอปพลิเคชันได้ทันที

- พิมพ์คำสั่ง `npx create-next-app@latest` เพื่อสร้าง Next.js แอปพลิเคชันด้วยเวอร์ชันล่าสุด

```

Command Prompt
C:\Users\jeera\next_dev>npx create-next-app@latest
  
```

- ในการสร้างโปรเจกต์ครั้งแรก จะต้องยืนยันคำสั่ง โดยกรอก `y` แล้วกดปุ่ม `<Enter>`

```

npm exec create-next-app@latest
C:\Users\jeera>npx create-next-app@latest
Need to install the following packages:
create-next-app@13.4.12
Ok to proceed? (y) y
  
```

- ตั้งชื่อโฟลเดอร์สำหรับเก็บข้อมูลโปรเจกต์ สำหรับในตัวอย่างตั้งชื่อเป็น `first-app` จากนั้นกดปุ่ม `<Enter>`

```

C:\Windows\system32\cmd.exe
C:\Users\jeera>npx create-next-app@latest
Need to install the following packages:
create-next-app@13.4.12
Ok to proceed? (y) y
? What is your project named? » first-app
  
```

- ที่ TypeScript ให้เลือกเป็น NO เนื่องจากเราจะใช้ JavaScript ในการเขียนโค้ดเป็นหลัก (ไม่ต้องการใช้ TypeScript)
- ที่ ESLint ให้เลือก YES เพื่อเลือกใช้งาน ESLint ซึ่งเป็นเครื่องมือสำหรับช่วยตรวจสอบโค้ด JavaScript (หรือ TypeScript) พร้อมทั้งแนะนำวิธีเขียนโค้ดที่ถูกต้องมาให้
- ที่ Tailwind CSS ให้เลือกเป็น YES เนื่องจากต้องการปรับแต่งสไตล์ หรือหน้าตาของแอปพลิเคชันโดยใช้ Tailwind CSS
- ที่ src ให้เลือกเป็น NO เนื่องจากไม่ต้องการกำหนดเนื้อหาของแอปพลิเคชันไว้ภายในโฟลเดอร์ src
- ที่ App Router ให้เลือกเป็น YES เนื่องจากต้องการใช้คุณสมบัติ App Router ที่เป็นคุณสมบัติใหม่ของ Next.js
- ในส่วนของการตั้งค่าเกี่ยวกับ import alias ให้เลือก NO เนื่องจากไม่ต้องการเปลี่ยนแปลงการตั้งค่าเกี่ยวกับการอ้างอิงไฟล์

```

C:\Users\jeera>npx create-next-app@latest
Need to install the following packages:
create-next-app@13.4.12
Ok to proceed? (y) y
✓ What is your project named? ... first-app
✓ Would you like to use TypeScript? ... No / Yes
✓ Would you like to use ESLint? ... No / Yes
✓ Would you like to use Tailwind CSS? ... No / Yes
✓ Would you like to use `src/` directory? ... No / Yes
✓ Would you like to use App Router? (recommended) ... No / Yes
✓ Would you like to customize the default import alias? ... No / Yes
Creating a new Next.js app in C:\Users\jeera\first-app.

```

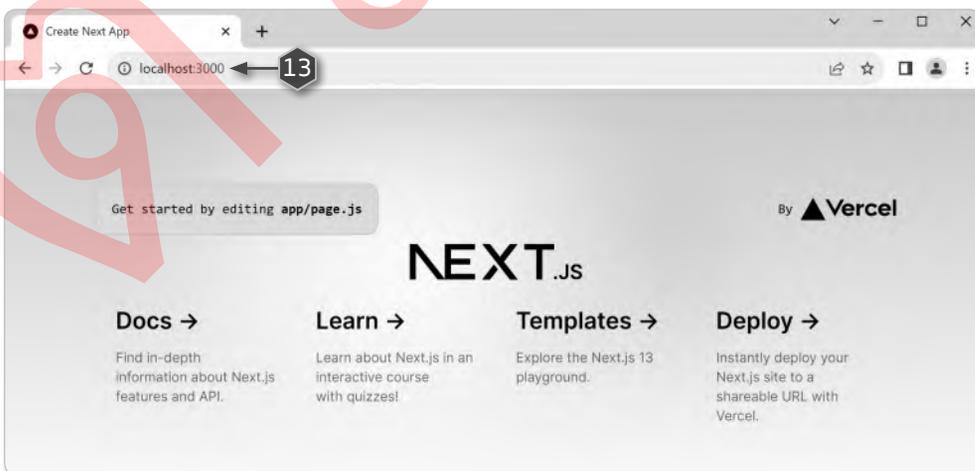
10. เข้าไปยังโฟลเดอร์ first-app โดยพิมพ์คำสั่ง cd first-app
11. ทดลองรัน Next.js แอปพลิเคชัน ในโหมดนักพัฒนา (development) โดยพิมพ์คำสั่ง npm run dev
12. ให้สังเกต URL และหมายเลขพอร์ตที่ใช้กับแอปพลิเคชัน (ในตัวอย่างคือ http://localhost:3000)

```

C:\Users\jeera>cd first-app
C:\Users\jeera\first-app>npm run dev
> first-app@0.1.0 dev
> next dev
- ready started server on 0.0.0.0:3000, url: http://localhost:3000
Attention: Next.js now collects completely anonymous telemetry regarding usage.
This information is used to shape Next.js' roadmap and prioritize features.
You can learn more, including how to opt-out if you'd not like to participate
in this anonymous program, by visiting the following URL:

```

13. เปิดเบราว์เซอร์ (เช่น Google Chrome) จากนั้นกรอก URL เป็น http://localhost:3000 จะพบ Next.js แอปพลิเคชันถูกแสดงบนเบราว์เซอร์ตามต้องการ

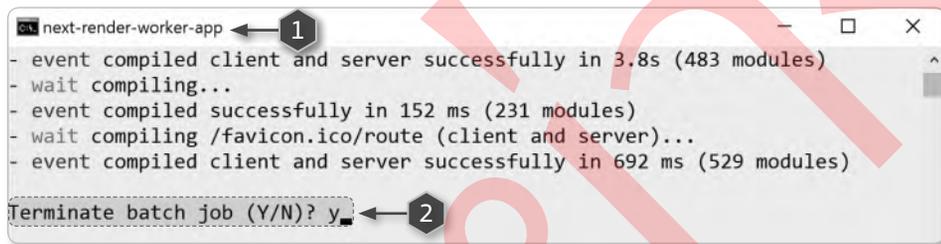


หยุดการทำงานของเว็บเซิร์ฟเวอร์

การใช้คำสั่ง `npm run dev` เพื่อรัน Next.js แอปพลิเคชัน จะมีการกำหนดสภาพแวดล้อมเพื่อจำลองการทำงานของเว็บเซิร์ฟเวอร์ ทำให้นักพัฒนาสามารถทดสอบ Next.js จากคอมพิวเตอร์ส่วนบุคคลได้ โดยไม่จำเป็นต้องฝากไว้กับเว็บเซิร์ฟเวอร์จริงที่อยู่ในอินเทอร์เน็ต

เมื่อใดที่ต้องการยกเลิกการทดสอบแอปพลิเคชัน ก็ให้ปิดระบบจำลองเว็บเซิร์ฟเวอร์ ด้วยการกดปุ่ม `<Ctrl+C>`

1. ไปยังหน้าต่าง Terminal (หรือ Command Prompt) ที่ได้สั่งรันแอปพลิเคชันเอาไว้ (หน้าต่างที่ใช้คำสั่ง `npm run dev`) กดปุ่ม `<Ctrl+C>` เพื่อสั่งปิดการทำงานของเซิร์ฟเวอร์
2. กรอก `y` แล้วกดปุ่ม `<Enter>`



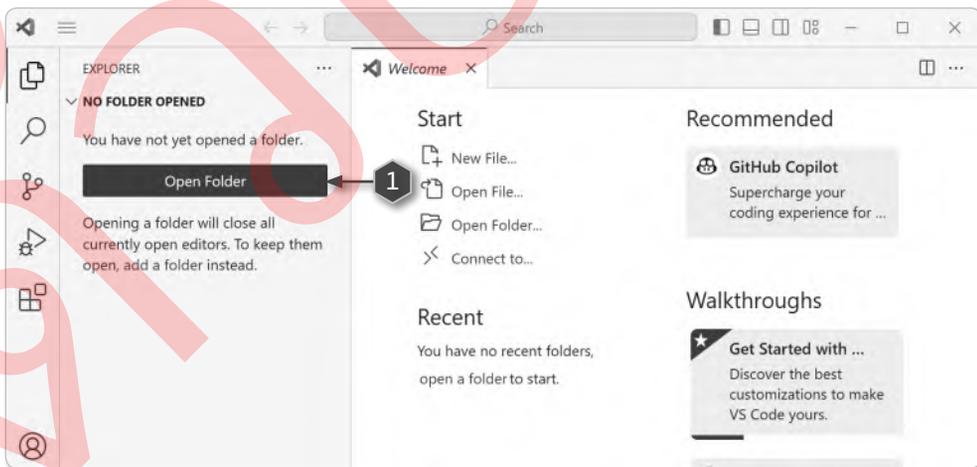
```

next-render-worker-app
- event compiled client and server successfully in 3.8s (483 modules)
- wait compiling...
- event compiled successfully in 152 ms (231 modules)
- wait compiling /favicon.ico/route (client and server)...
- event compiled client and server successfully in 692 ms (529 modules)
Terminate batch job (Y/N)? y
  
```

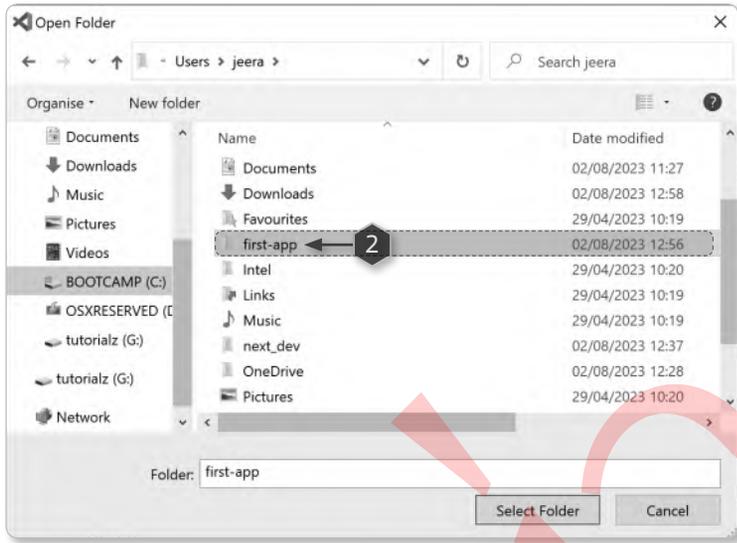
เปิด Next.js โปรเจกต์มาแก้ไข

หลังจากได้สร้าง Next.js แอปพลิเคชันไว้ในโฟลเดอร์ `first-app` เรียบร้อยแล้ว หากต้องการแก้ไขโค้ดภายในโปรเจกต์ก็สามารถใช้ VS Code เปิดโปรเจกต์ขึ้นมาแก้ไขได้ ดังนี้

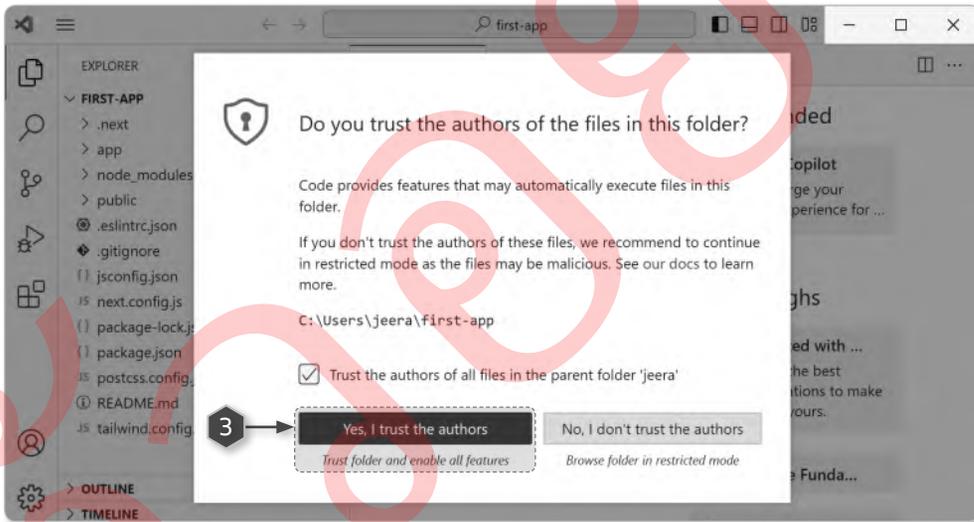
1. เปิดโปรแกรม Visual Studio Code จากนั้นให้คลิกที่ `Open Folder` เพื่อเปิดโฟลเดอร์



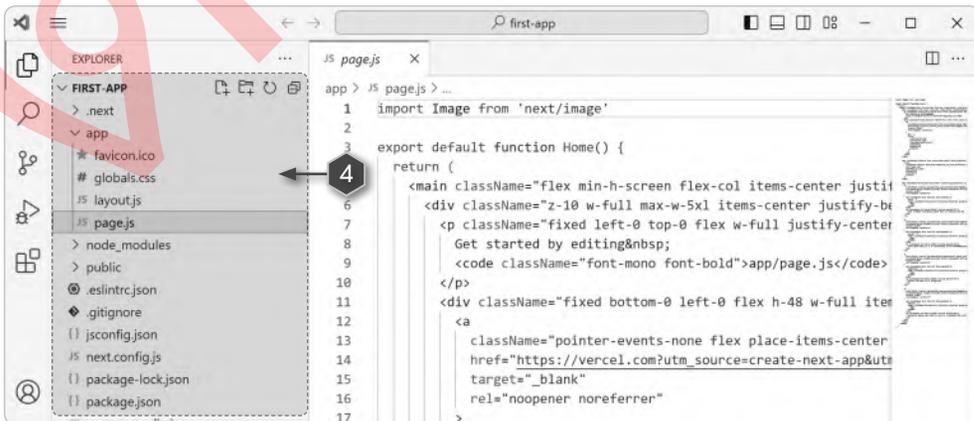
2. คลิกเลือกโฟลเดอร์ที่ต้องการ สำหรับในตัวอย่างนี้ได้เลือกโฟลเดอร์ `first-app` ซึ่งเป็นโฟลเดอร์ที่เก็บ Next.js แอปพลิเคชันเอาไว้



3. การใช้งานครั้งแรกอาจมีหน้าต่างเพื่อขออนุญาตเข้าถึงไฟล์ต่าง ๆ ในโฟลเดอร์ ให้คลิกปุ่ม Yes

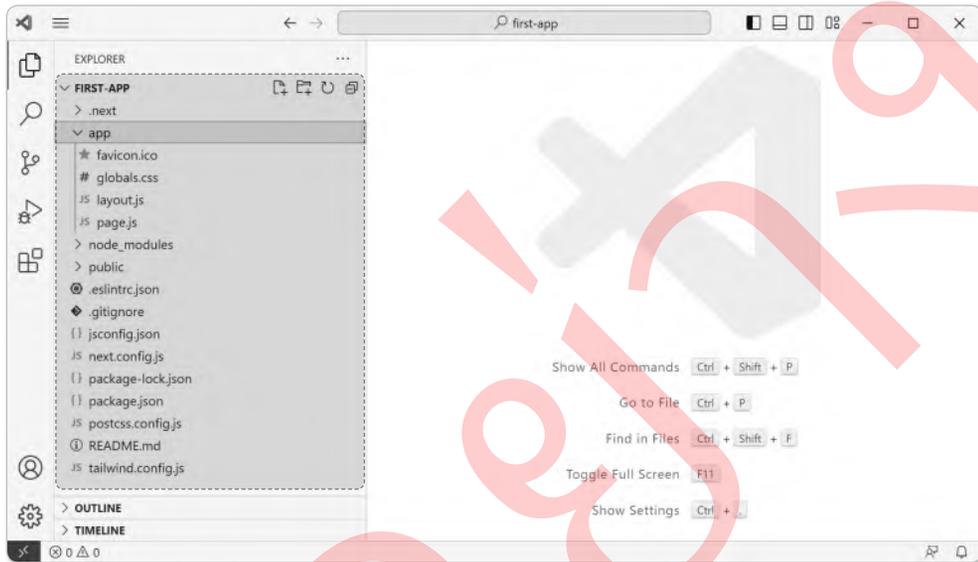


4. จะพบโฟลเดอร์และไฟล์ต่าง ๆ ที่อยู่ในโปรเจกต์



ไฟล์พื้นฐานภายในโปรเจกต์ของ Next.js

เนื่องจาก Next.js เป็นเฟรมเวิร์คที่มีการกำหนดกฎเกณฑ์บางอย่างเอาไว้ ดังนั้น การกำหนดไฟล์เตอร์ หรือ การกำหนดชื่อไฟล์จะต้องเป็นไปตามเงื่อนไขที่ Next.js กำหนด เช่น ไฟล์ page.js ที่อยู่ในไฟล์เตอร์ app ใช้เพื่อกำหนดหน้าโฮมเพจ ส่วนไฟล์ layout.js ใช้กำหนดเค้าโครงของหน้าเว็บเพจ เป็นต้น



ต่อไปนี้เป็นไฟล์เตอร์และไฟล์สำคัญของ Next.js ที่จำเป็นต้องทราบ

📁 ไฟล์เตอร์ app

ในขณะติดตั้ง Next.js หากเลือกใช้ App Router หน้าเว็บเพจต่าง ๆ ของเว็บไซต์จะถูกกำหนดไว้ภายในไฟล์เตอร์ app โดยชื่อไฟล์เตอร์ที่อยู่ในไฟล์เตอร์ app จะถูกนำไปใช้เป็นชื่อ path ของ URL เช่น การกำหนดไฟล์ /app/about/page.js จะเป็นการกำหนดพาร ‘/about’ ให้กับเว็บไซต์โดยอัตโนมัติ เช่น www.yourwebsite.com/about

📄 ไฟล์ /app/page.js

เมื่อเลือกใช้คุณสมบัติ App Router ไฟล์ page.js ที่อยู่ในไฟล์เตอร์ app จะถูกกำหนดให้เป็นหน้าโฮมเพจโดยอัตโนมัติ

📄 ไฟล์ /app/layout.js

เมื่อมีการเลือกใช้คุณสมบัติ App Router ไฟล์ layout.js ที่อยู่ในไฟล์เตอร์ app จะถูกกำหนดให้เป็นเค้าโครงหลักของแอปพลิเคชัน (Root Layout) เช่น กำหนดให้เมนูหลักอยู่ด้านบน รูปภาพแสดงอยู่ด้านขวา ส่วนเนื้อหาของเว็บเพจอยู่ด้านซ้าย เป็นต้น

ใช้ Next.js ร่วมกับ ChatGPT

สร้างได้ทั้ง Frontend และ Backend

ด้วยเนื้อหาที่แน่นแต่กระชับ
ปรับแต่งให้เข้าใจง่าย

อ่านและทำตามได้แบบ
Step-By-Step
เก่งได้ แม้ไม่มีพื้นฐานมาก่อน



- ทบทวนพื้นฐาน React ที่จำเป็นต้องทราบสำหรับใช้งานกับ Next.js
- การใช้ ChatGPT ช่วยพัฒนาโปรแกรม เช่น กำหนดโครงสร้าง เขียนโค้ด แก้ไขโค้ด วิเคราะห์ข้อมูล ฯลฯ
- การใช้ Tailwind CSS ปรับแต่งสไตล์ให้กับแอปพลิเคชัน
- อธิบายความแตกต่างระหว่าง Client Component และ Server Component
- สร้างเว็บเพจโดยใช้ Server-Side Rendering และ Static Site Generation
- พื้นฐานการใช้งาน App Router ทั้งในฝั่ง Frontend และ Backend
- การใช้งาน Server Actions ส่งข้อมูลจากแบบฟอร์ม

- แนะนำวิธีสร้าง API การโหลดข้อมูลจาก API และวิธีเก็บข้อมูลลงแคช
- การกำหนด Middleware สำหรับใช้จัดการกับ HTTP Request
- วิธีกำหนด Metadata และวิธีจัดการรูปภาพ สำหรับการทำ SEO
- การใช้ Next.js ร่วมกับฐานข้อมูล (MongoDB และ Firebase Firestore Database)
- สร้างระบบยืนยันตัวตนโดยใช้ Next-Auth และ Firebase Authentication



ISBN (eBook) 885-909-931-023-9



8 859099 310239

ราคา 650 บาท



ซื้อสะดวก ส่งถึงบ้านที่ Shopee และ Lazada หรือผ่านทาง
ร้านหนังสือออนไลน์ www.thinkbeyondbook.com



thinkbeyond books