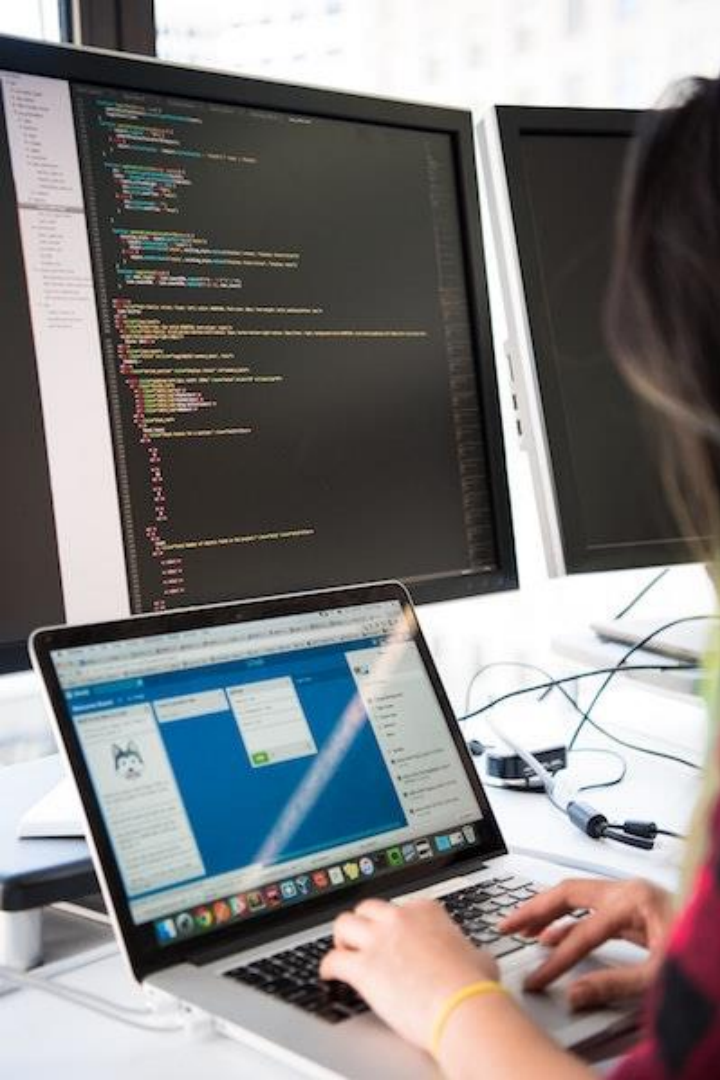


Important Instructions

- การส่ง Assignment ผ่าน [Github Classroom](#)
 - งานใน **Week 07** เป็นงานต่อจาก **Assignment** ใน **Week 06** แต่ใช้ Repository เริ่มต้นที่ให้มา ผ่าน Assignment Invitation Link สำหรับ Assignment07
 - การส่ง Assignment ภายใต้ Student Identity ที่กำหนดไว้ตั้งแต่ตอนรับ Invitation ของ Assignment 01
- นิสิตทดสอบ
 - Assignment มาพร้อม Test Script สำหรับทดสอบ project ให้นิสิตทดสอบตาม instruction ในสไลด์ชุดนี้
- นิสิตส่งงานโดย
 - Deploy เว็บไซต์ที่สร้างจาก Assignment ขึ้น Vercel
 - Commit และ push ขึ้น repository ที่ได้จาก Assignment
 - นิสิตต้องใส่ link ไปยัง Vercel website ของนิสิต ใน **README.md** ใน Repository

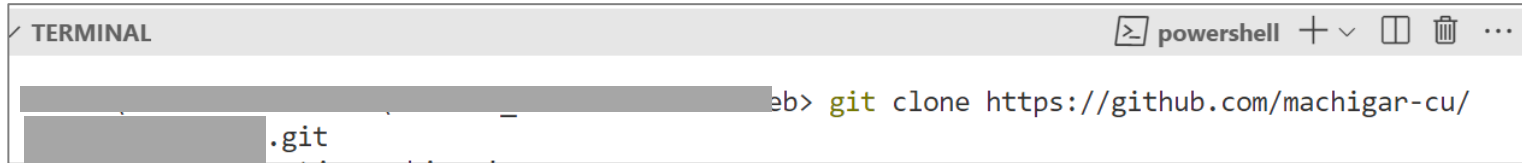


Assignment 7: Router & Navigation

Prepare Class Environment



- เปิด Folder สำหรับเก็บหน้า React Web ที่สร้างไว้ใน Assignment 03 ด้วย Visual Studio Code
- ไปที่ Menu **View > Terminal**
 - Clone Repo ของตนเอง พิมพ์คำสั่ง `git clone REPO_URL_FROM_GITHUB`



```
TERMINAL powershell + - [x] [x] ...
C:\Users\user> git clone https://github.com/machigar-cu/
.git
```

Note: ถ้ายังไม่ได้ setup git credential ในเครื่อง อาจขึ้นว่า repository not found ให้เพิ่ม user ของ git account ตามด้วย @ ไปด้านหน้า github
`git clone https://YOUR_USER@github.com/machigar-cu/.....git`

- **YOUR_USER** คือ Github user ของนิสิต

- เปิด Folder ที่ Clone มาได้ (`a07-swdevprac2-xxxxxx`)
ใน Visual Studio Code
 - จะพบไฟล์ตั้งต้นของ project

Prepare Class Environment



- ไปที่เมนู View > Terminal

- Configure git username และ email สำหรับ local repository

```
git config --local user.email "example@email.com"
```

```
git config --local user.name "your_userName"
```

```
machigar> git config --local user.email "[redacted]@gmail.com"  
machigar> git config --local user.name "machigar"
```

- ติดตั้ง node packages `npm install`
- Copy ไฟล์เตอร์ `public` และ `src` จาก project ของ Assignment 06 มาทับ ไฟล์เตอร์ `public` และ `src` ใน project ของ Assignment 07 นี้
- ลอง run project จะได้ project ที่ทำไว้ใน Assignment 06 `npm run dev`

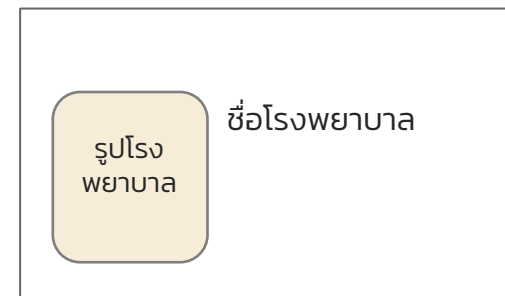


ใน Assignment 7 นี้ จะทำการย้ายการแสดงผลข้อมูลโรงพยาบาล จากหน้าแรก มาไว้ที่ route `\hospital`

- สร้าง Route Group (`hospitalinfo`) ไว้ใน `app` พร้อมสร้าง route `\hospital` ใน route group นี้ โดยมีหน้าเพจ `page.tsx` สำหรับ route นี้
- ย้าย `CardPanel` ที่สร้างไว้ใน Assignment 06 จากหน้าแรก มาแสดงบนหน้า `page` ของ route ใหม่
- ปรับปรุง `CardPanel.tsx` เพื่อจำลองการดึงข้อมูล โดยปรับเปลี่ยนข้อมูลที่เคยนำมาแสดงบน card แบบ static ให้เป็น
 - Mock ข้อมูลรายการโรงพยาบาล โดยเก็บไว้ใน array ของ Object ที่มีโครงสร้างที่เหมาะสม ทั้งนี้ ให้แต่ละโรงพยาบาลมี `hid` เป็น ID ของโรงพยาบาลดังนี้
 - Chulalongkorn Hospital มี `hid` เป็น 001
 - Rajavithi Hospital มี `hid` เป็น 002
 - Thammasat University Hospital มี `hid` เป็น 003
 - ใช้ loop ในการดึงข้อมูลมาวางบน card แต่ละอัน ลักษณะเดียวกับงานที่เรียนในชั้นเรียน



- ใน Route Group `(hospitalinfo)\hospital` สร้าง Dynamic Segment `\hospital\[hid]` โดยมีหน้าเพจ `page.tsx` สำหรับ route segment นี้
 - สำหรับแสดงรายละเอียดของโรงพยาบาลที่มี `hid` เป็นค่านั้น ๆ
 - เช่น user navigate มาที่ `/hospital/001` จะแสดงรายละเอียดของโรงพยาบาลที่มี id เป็น 001
 - ข้อมูลรายละเอียดของโรงพยาบาลให้ใช้ Mock ข้อมูลโรงพยาบาลใน Map ของ Object โดยมี `hid` เป็น key ลักษณะเดียวกับงานที่เรียนในชั้นเรียน
 - เมื่อคลิก card ที่แสดงโรงพยาบาลที่มี id ใดใน `CardPanel` ที่ `/hospital` จะ navigate มาที่ `/hospital/hid` เมื่อ `hid` เท่ากับ id นั้น ๆ โดยหน้าแสดงรายละเอียดให้แสดงรูปและชื่อของโรงพยาบาล
 - ใช้ Link component ในการทำ navigation





- ปรับปรุง `Banner.tsx` ซึ่งเป็น Banner ที่วางบนหน้าแรกของ app ให้
 - รองรับการเปลี่ยนรูปภาพของ Banner เมื่อคลิกลงบน Banner โดยการใช้ `useState`
 - โดยมีภาพ Banner รวม 4 ภาพ และเมื่อคลิกภาพจะเปลี่ยนเป็นภาพถัดไป เมื่อครบแล้วจะวนกลับมาภาพแรกไฟล์ภาพเป็น
 - `/img/cover.jpg`
 - `/img/cover2.jpg`
 - `/img/cover3.jpg`
 - `/img/cover4.jpg`ให้ `/img/cover.jpg` เป็นภาพตั้งต้น
 - เลือก ใช้ภาพที่ไม่พิดลิขสิทธิ์
 - สร้างปุ่มที่มุมขวาล่างของ banner แสดงข้อความ `Select Hospital`
 - เมื่อคลิกปุ่มให้ `navigate` ไป route `/hospital` โดยใช้ `useRouter` ในการทำ navigation

Assignment test cases

Assignment 07 มี 4 test cases (รวม Autograde 5 คะแนน)

- ลอง run test case ที่เครื่องของตนเอง ไปที่เมนู **View > Terminal**
 - Run **test case 1 (1 คะแนน)** ด้วยคำสั่ง
`npm run test -- case1.test.tsx`
 - Run **test case 2 (2 คะแนน)** ด้วยคำสั่ง
`npm run test -- case2.test.tsx`
 - Run **test case 3 (1 คะแนน)** ด้วยคำสั่ง
`npm run test -- case3.test.tsx`
 - Run **test case 4 (1 คะแนน)** ด้วยคำสั่ง
`npm run test -- case4.test.tsx`

Development & Deployment

- ใช้แนวทางการ Deploy แบบเดียวกับใน Assignment A03 ถึง A06
 - สร้าง Repository บน Personal GitHub Account สำหรับ Assignment A07
 - ดู Instructions ใน Assignment A01
 - Config ให้ local project มี multiple remote repositories และ Push งานขึ้น repositories ทั้งสอง
 - ดู Instructions ใน Assignment A01
 - Deploy project จาก Personal GitHub Account ขึ้น Vercel
 - ดู Instructions ใน Assignment A01
 - Configure และ Deploy บน Vercel โดยระบบจะตรวจจับได้ว่าเป็น Next.js Framework
- ใส่ link ไปยัง Vercel website ของนิสิต ใน **README.md** ใน Repository