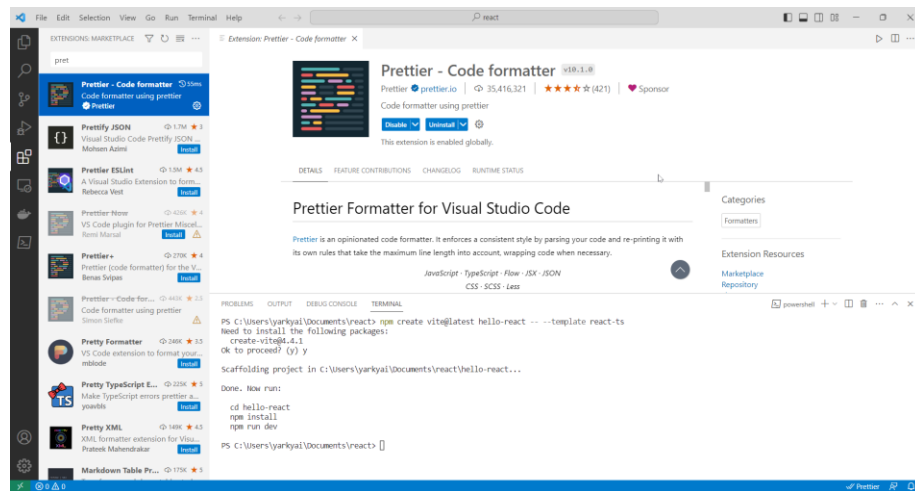


เขียนเว็บด้วย React + TypeScript

สิ่งที่ต้องมี

- VS Code - ตัว Text Editor สำหรับเขียนโค้ด (หรือใช้ IDE/Editor ที่ชอบก็ได้)
- Node.js / NPM - ติดตั้ง Node.js และ NPM เพื่อให้สามารถติดตั้ง Library ต่างๆได้
- Terminal - มี Command Line Tool เช่น Terminal, iTerm2, Power Shell หรือสามารถใช้ built-in Terminal ที่มากับ VS Code ได้
- Prettier - เพื่อเอาไว้ format code ของเราให้เป็นแนวทางเดียวกัน (สามารถใช้ Prettier Extension VS Code ได้)



- มีความเข้าใจ HTML/CSS/JS เบื้องต้น หรือเคยเขียนเว็บไซต์มาก่อน

1. เริ่มสร้างโปรเจก

copy คำสั่งตัวนี้ไปรันครับ พิมพ์เองเดี๋ยวผิด (เราจะใช้ Vite ในการสร้างโปรเจก React ขึ้นมา ซึ่ง Vite เป็น Frontend Tooling ที่ช่วยเรื่อง Config ต่างๆ ทำให้เราไม่ต้องเสียเวลาตรงนี้)

```
npm create vite@latest hello-react -- --template react-ts
```

```
PROBLEMS  OUTPUT  DEBUG CONSOLE  TERMINAL
PS C:\Users\yarkyai\Documents\react> npm create vite@latest hello-react -- --template react-ts
Need to install the following packages:
  create-vite@4.4.1
ok to proceed? (y) y

Scaffolding project in C:\Users\yarkyai\Documents\react\hello-react...

Done. Now run:

  cd hello-react
  npm install
  npm run dev

PS C:\Users\yarkyai\Documents\react> I
```

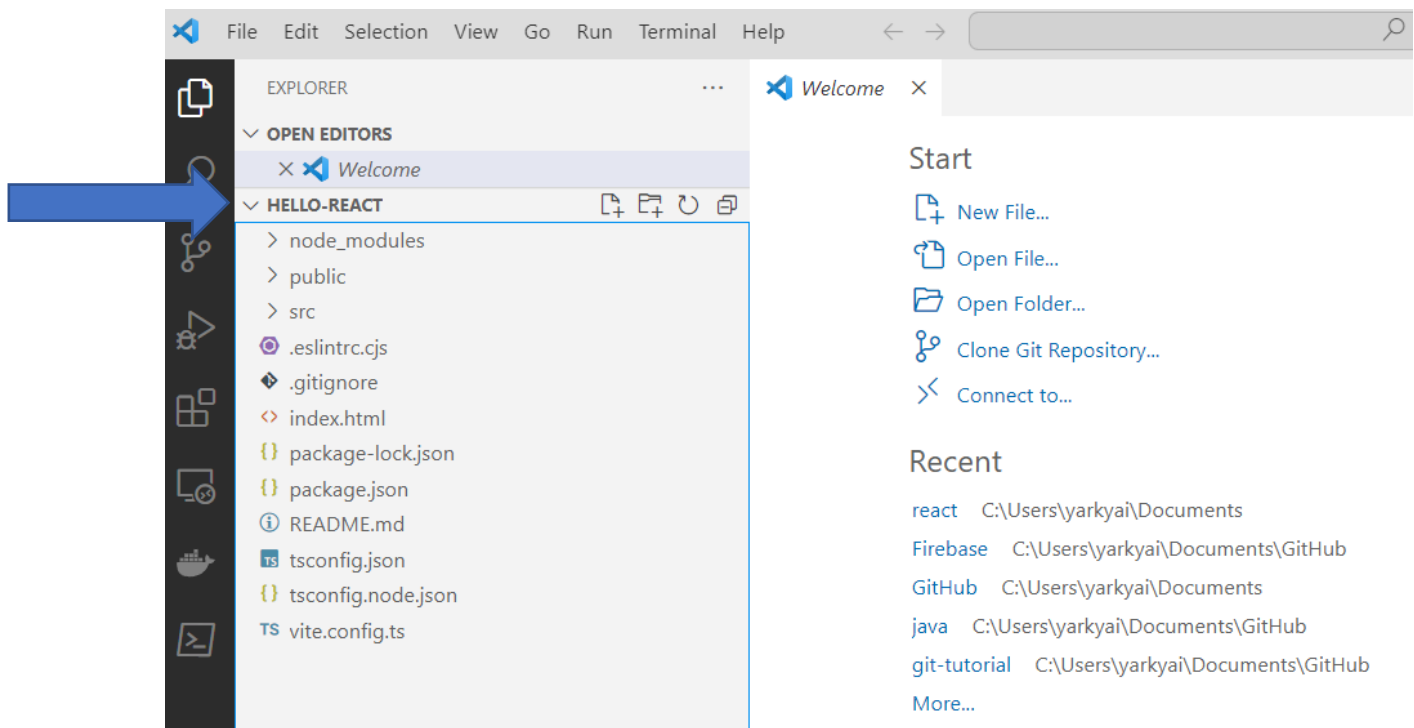
หลังจากรันคำสั่งแล้ว เราจะได้โครงสร้างใน VS code ตามนี้ครับ

- src - เป็น folder หลักที่เราจะเขียน React กันที่ folder นี้
- index.html - เป็นไฟล์ html หลักของเรา
- package.json - เป็นไฟล์ที่บอกว่าโปรเจกต์เรา ชื่ออะไร มี dependencies อะไรบ้าง
- tsconfig.json - เป็น TypeScript config (เรายังไม่ต้องสนใจ ใช้เป็น default ก่อนครับ)

จากนั้นเราจะเริ่ม start dev server ครับ

```
PS C:\Users\yarkyai\Documents\react> cd hello-react
PS C:\Users\yarkyai\Documents\react\hello-react> code .
PS C:\Users\yarkyai\Documents\react\hello-react> █
```

หลังจากพิมพ์ code . VS code จะเปิดหน้าต่างขึ้นมาไม่ต้องตกใจครับ



พิมพ์คำสั่ง npm install (ติดตั้ง dependencies)

```
PS C:\Users\yarkyai\Documents\react\hello-react> npm install

added 201 packages, and audited 202 packages in 17s

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

found 0 vulnerabilities
```

จากนั้นเราจะ Start dev server ด้วยคำสั่ง npm run dev

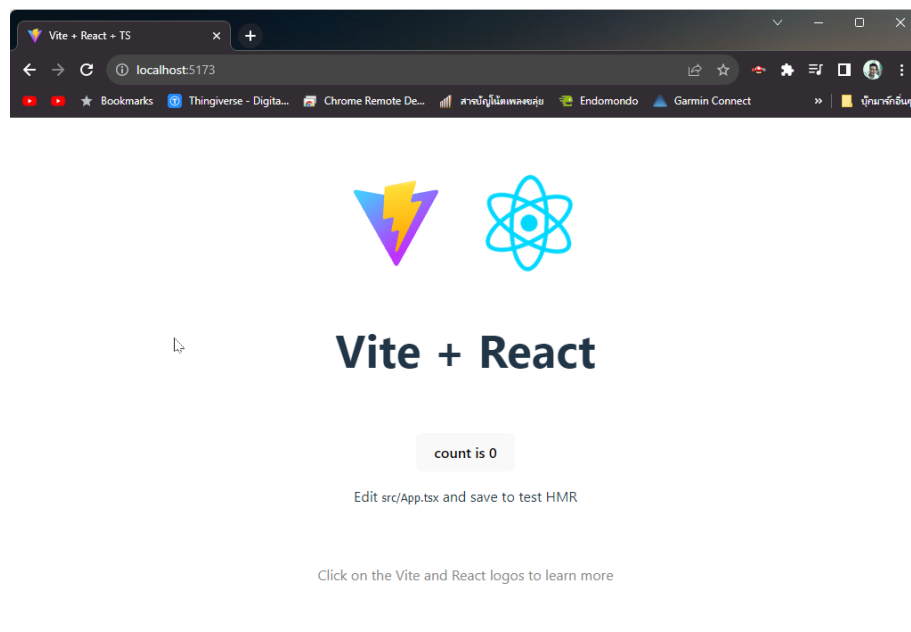
```
PS C:\Users\yarkyai\Documents\react\hello-react> npm run dev

> hello-react@0.0.0 dev
> vite

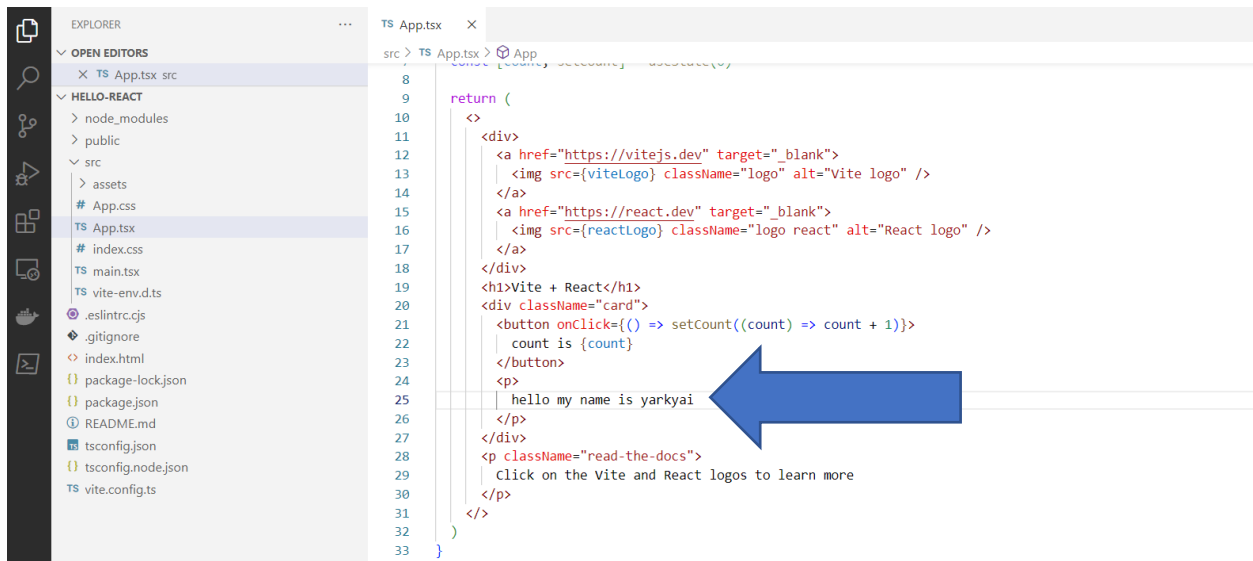
VITE v4.4.9 ready in 612 ms

➔ Local:   http://localhost:5173/
➔ Network: use --host to expose
➔ press h to show help
```

จะเห็นว่า Local host เราทำงานที่ port 5173 แล้วครับ ทีนี้ก็ไปเปิดเว็บที่ <http://localhost:5173> ดูครับ



ที่นี้เรามาลองแก้ไขไฟล์กันดีกว่าครับ ไปที่ src\App.tsx แล้วลองเปลี่ยน code ดู ตัวอย่างผมเปลี่ยนบรรทัดที่ 25 เป็น hello my name is yarkyai



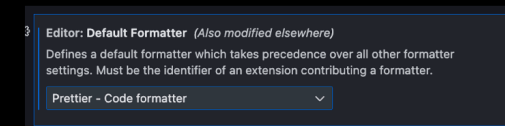
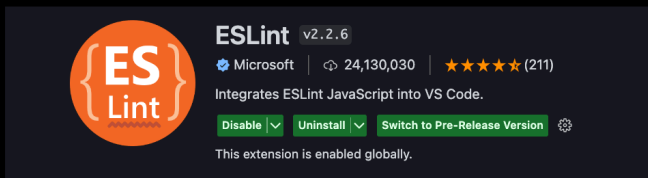
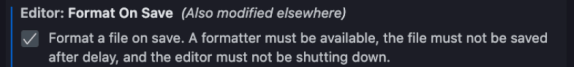
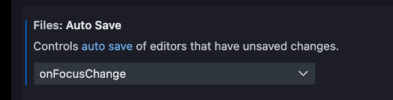
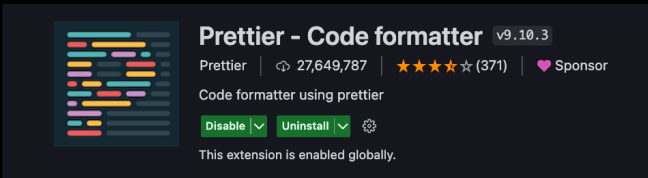
```
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
--
return (
  <div>
    <a href="https://vitejs.dev" target="_blank">
      <img src={viteLogo} className="logo" alt="Vite logo" />
    </a>
    <a href="https://react.dev" target="_blank">
      <img src={reactLogo} className="logo react" alt="React logo" />
    </a>
  </div>
  <h1>Vite + React</h1>
  <div className="card">
    <button onClick={() => setCount((count) => count + 1)}>
      count is {count}
    </button>
    <p>
      hello my name is yarkyai
    </p>
  </div>
  <p className="read-the-docs">
    Click on the Vite and React logos to learn more
  </p>
</>
)
```

เสร็จแล้วให้กด save จะเห็นว่าหน้าเพจของเราจะเปลี่ยนทันทีโดยที่เราไม่ต้อง refresh เลยยยย ดิงม แต่ดิงมเฉพาะเราละ อย่าเอาไปทำบนเพจจริง ไม่งั้นเราลองอะไรเข้าไปหน้าที่เราเขียนออกสู่สาธารณะเลยนะไว้ยัย อย่างเขียนว่า แมนอยู่อ่อนอะไรแบบเนี่ยะ อย่าหาทำ

Auto save และ Auto formatting

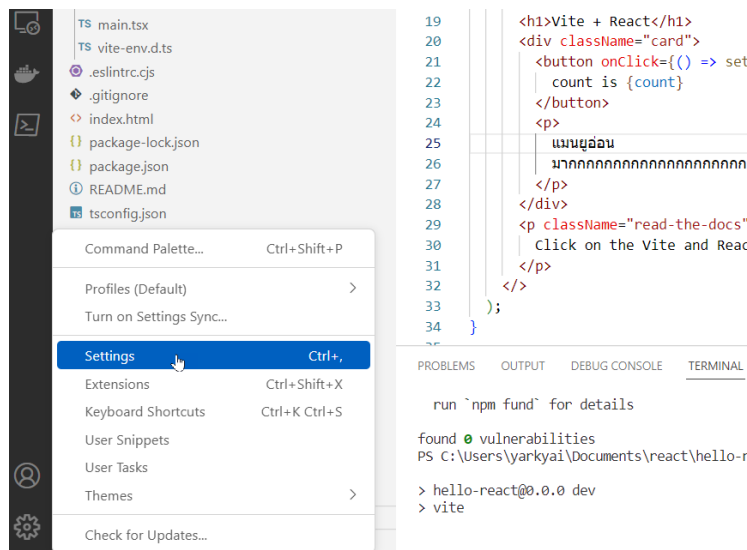
สำหรับคนที่ต้องการ Auto Save และ Auto Formatting สามารถทำได้ตามนี้นะครับ

Extension & Settings

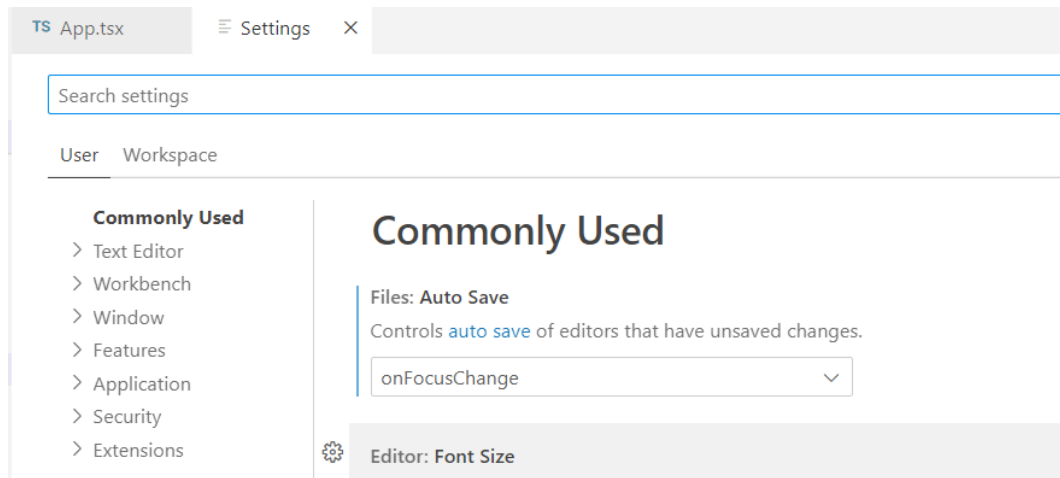


- Prettier - เป็น Extensions หลัก ที่ควรมีติดไว้
- ES Lint - เป็นตัวช่วยตรวจสอบ code ของเรา (Optional)

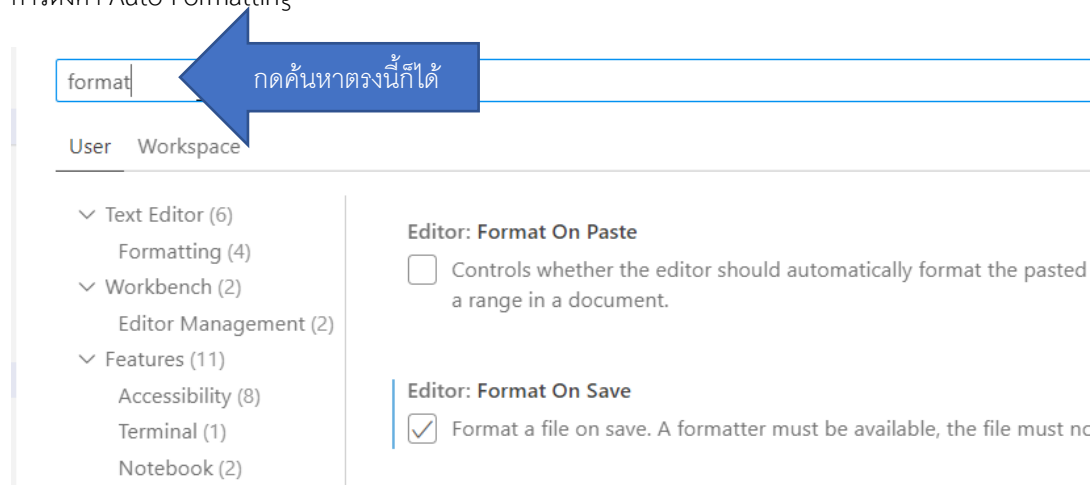
มาดูการตั้งค่า ทั้ง 3 ค่า ที่แนะนำ ก่อนอื่นไปที่ settings ใน VS code



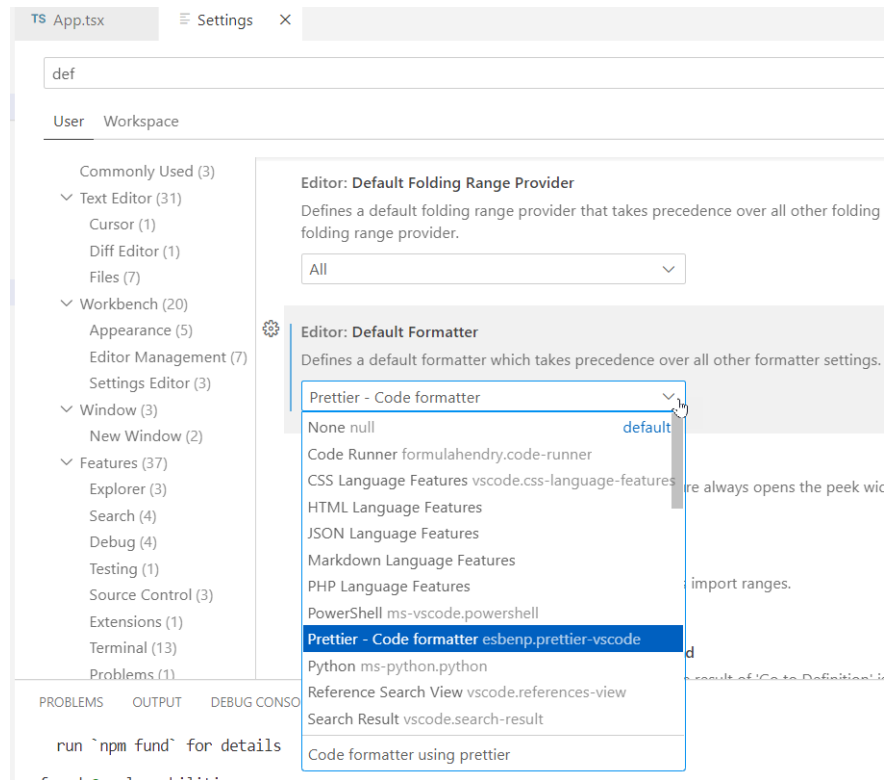
Auto Save เลือก onFocusChange



การตั้งค่า Auto Formatting



การเลือก Default Formatting



ทดลองพิมพ์ Code แล้วกดแล้วเอาเมาท์ ไปกดที่ terminal โปรแกรมจะทำการ save code ให้อัตโนมัติ หรือทดลองพิมพ์ `<p> mahanakorn </p>` โดยให้ขีดซ้ายสุดเลย กด save จะเห็นว่า code จะด้งกลับมาสวยๆ ให้

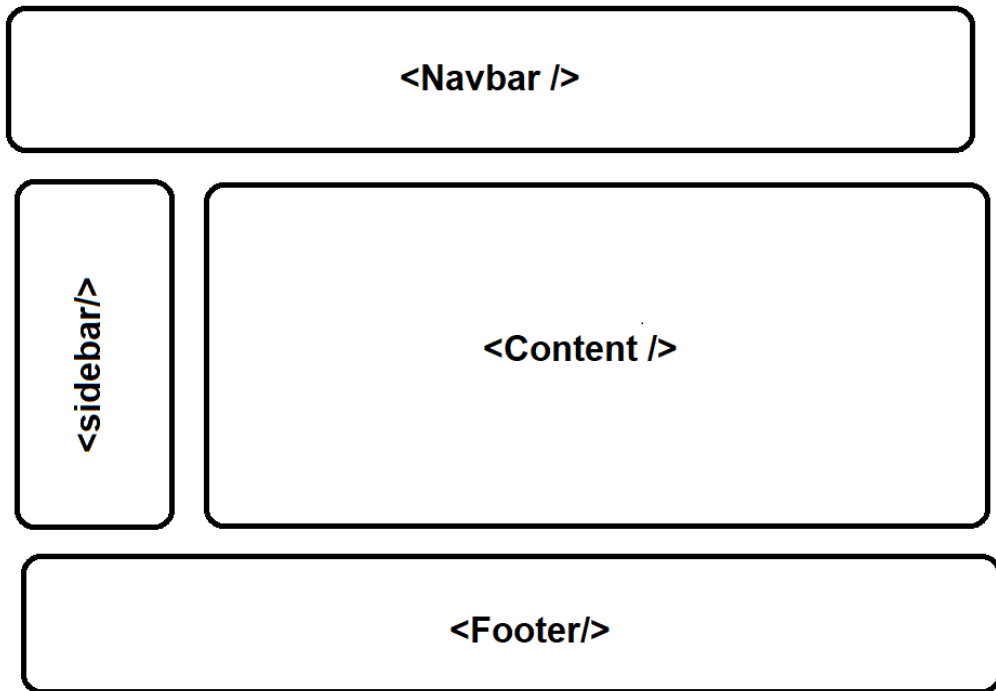
2. Basic React

React คือ JavaScript Library ที่ใช้สำหรับ ทำ User Interfaces (UI) เป็นโอเพ่นซอร์สและถูกพัฒนาขึ้นจากทีม Facebook

แนวคิดของ React คือ

- แบ่งเว็บออกเป็นส่วนย่อยๆ เราเรียกมันว่า Component
- Component ช่วยให้เราไม่ต้องเขียนโค้ดซ้ำๆ
- ในช่วงแรกๆ เริ่มมาจากแนวคิด Single Page Application (SPA)

- JSX ที่มีความคล้ายกับ HTML แต่มันคือ JavaScript ที่มีความสามารถเพิ่มขึ้น



การสร้าง Component

```
function MyButton() {

  return <button>This is my button component</button>

}
```

- Component คือ UI ที่มี logic ภายใน หรือเป็นแค่ markup อย่างเดียวก็น่าได้
- เป็น function ธรรมดา ที่ return JSX
- เราจะสร้าง Component โดยแบ่งเป็นส่วนเล็กๆ แค่ Button หรือ Link หรือจะสร้าง Component 1 อัน ให้ render ทั้งหน้า ก็ได้ (อยู่ที่เราออกแบบ)
- แม้จะเป็น javascript function แต่ naming convention แนะนำให้ตั้งชื่อเป็น capital letter คือ เริ่มต้นด้วยตัวพิมพ์ใหญ่ เช่น `MyButton` , `Button` , `Image` (จะได้ไม่ซ้ำกับ tag html ที่มีอยู่แล้ว)

- เวลาที่ เราจะ Return element ใน Component เราจะใช้ `()` กรณีมีหลายบรรทัด

การเรียกใช้งาน Component เหมือนกับ การใช้งาน tag html ปกติเลย

```
<MyComponent></MyComponent>
```

เพียงแค่นี้ใน React เมื่อเราไม่มี content ใน element เราสามารถใช้ closing tag ได้เลยแบบนี้

```
<MyComponent />
```

ทดสอบการทำงานลองเขียนข้อความ (ไปก๊อปปามาจากไหนก็ได้แล้วจัด format เหมือนตัวอย่าง)

```
19 <h1>Vite + React</h1>
20 <div className="card">
21   <button onClick={() => setCount((count) => count + 1)}>
22     count is {count}
23   </button>
24   <p>
25     เติ่งแชมป์เท่าไร? บอนรับพนันปรับ ลิเวอร์พูล & แมนฯ ยูไนเต็ด
26     โอกาสชิวแชมป์พรีเมียร์ลีก
27   </p>
28   <p>
29     บริษัทรับพนันถูกกฎหมายของอังกฤษ มีการปรับเรตลดอัตราจ่ายการที่
30     อาร์เซนอล จะครองแชมป์ พรีเมียร์ลีก 2023/24 ลงเล็กน้อย
31     หลังจากทีมปืนใหญ่เป็นอีกรายที่เดิหน้าชนะ 2 เกมซ้อน ในขณะที่ เชลซี
32     ถูกเพิ่มเรตจนกลายเป็นเต็ง 5 ตามหลัง แมนเชสเตอร์ ยูไนเต็ด ไปแล้ว
33     พรีเมียร์ลีก ผ่านสัปดาห์ที่ 2 ไป โดยมี โบรน์ตัน, แมนเชสเตอร์ ซิตี้
34     และ อาร์เซนอล เป็นเพียง 3 ทีมที่กวาด 6 แต้มเต็มเข้ากระเป๋า
35   </p>
36   <p>
37     ในขณะที่ แมนฯ ยูไนเต็ด กับ เชลซี ต่างพลาดทำแพ้นัดแรกของตัวเองไปแล้ว
38     ในการปรับเรตล่าสุดของร้านพูลชั้นนำอย่าง สกาย เบต หลังจากเกม อาร์เซนอล
39     เบียดชนะ คริสตัล พาเลซ 1-0 เมื่อวันจันทร์
40     มีการปรับลดอัตราจ่ายแชมป์ของเต็ง 2 อาร์เซนอล ลงเล็กน้อย จาก 9/2 เป็น
41     4/1 อย่างไรก็ตาม เติ่งแชมป์ยังคงเป็น แมนฯ ซิตี้ แชมป์ 3 สมัยซ้อน
42     ตามเดิม ในเรต 2/3 ส่วนทาง เชลซี ถูกปรับเพิ่มอัตราจ่าย
43     ไปเป็นตัวเต็งลำดับที่ 5 ตามหลังทั้ง แมนฯ ยูไนเต็ด และ นิวคาสเซิล
44     ไปแล้ว หลังจากแพ้ว เวสต์แฮม เมื่อวันอาทิตย์
45   </p>
46 </div>
```

ลองดูที่หน้าเว็บ จะเห็นว่าข้อมูลก็เปลี่ยนตามที่เราได้ทำเอาไว้ แล้ว react มันทำอย่างไรล่ะ

ดูที่หน้า index.html จะเป็นว่ามีโปรแกรมไม่ก็บรรทัดเอง

ตามไปดูที่ไฟล์ main.tsx จะเห็นว่ามี render App เอาเป็นว่ามันเอาไฟล์ App.tsx มาทำงานนะ

ที่นี้เรามาลองสร้าง Component กันในไฟล์ App.tsx ให้ลองเขียน Function ตามรูปเลย

จากนั้นเราจะลองเรียกใช้ Function ของเราโดย เอาไปไว้ที่ตรงไหนก็ได้ที่เราต้องการให้แสดงข้อความใน Function แล้วสังเกตผลการทำงาน

```
40 | <p>
41 |   ในขณะที แมนฯ ยูไนเต็ด กับ เชลซี ต่างพลาดทำแฮตทริกของตัวเองไปแล้ว
42 |   ในการปรับเรตล่าสุดของร้านพลซันน่าอย่าง สกาย เบท หลังจากเกม อาร์เซนอล
43 |   เบียดชนะ คริสตัล พาเลซ 1-0 เมื่อวันจันทร์
44 |   มีการปรับลดอัตราจ่ายแชมป์ของเต็ง 2 อาร์เซนอล ลงเล็กน้อย จาก 9/2 เป็น
45 |   4/1 อย่างไรก็ตาม เต็งแชมป์ยังคงเป็น แมนฯ ซิตี้ แชมป์ 3 สมัยซ้อน
46 |   ตามเดิม ในเรต 2/3ส่วนทาง เชลซี ถูกปรับเพิ่มอัตราจ่าย
47 |   ไปเป็นตัวเต็งลำดับที่ 5 ตามหลังทั้ง แมนฯ ยูไนเต็ด และ นิวคาสเซิล
48 |   ไปแล้ว หลังจากแพ้ว เวสต์แฮม เมื่อวันอาทิตย์
49 | </p>
50 | <MyComponent></MyComponent>
51 | </div>
52 | <p className="read-the-docs">
53 |   Click on the Vite and React logos to learn more
54 | </p>
55 | </>
56 | );
57 | }
58 |
59 | export default App;
```

ทีนี้ถ้าเราต้องการย้าย บรรทัดที่ข้อความทั้งหมดไปไว้ใน Function ของเราละ ให้ทดลอง Cut ข้อความทั้ง 2 พารากราฟ ไปไว้ใน MyComponent

```
6 | function MyComponent() {
7 |
8 |   <p>
9 |     บริษัทรับพนันถูกกฎหมายของอังกฤษ มีการปรับเรตลดอัตราจ่ายการที่
10 |     อาร์เซนอล จะครองแชมป์ พรีเมียร์ลีก 2023/24 ลงเล็กน้อย
11 |     หลังจากทีมปืนใหญ่เป็นอีกรายที่เดินหน้าชนะ 2 เกมซ้อน ในขณะที่ เชลซี
12 |     ถูกเพิ่มเรตจนกลายเป็นเต็ง 5 ตามหลัง แมนเชสเตอร์ ยูไนเต็ด ไปแล้ว
13 |     พรีเมียร์ลีก ผ่านสัปดาห์ที่ 2 ไป โดยมี โบลตัน, แมนเชสเตอร์ ซิตี้
14 |     และ อาร์เซนอล เป็นเพียง 3 ทีมที่กวาด 6 แต้มเต็มเข้ากระเป๋า
15 |   </p>
16 |   <p>
17 |     ในขณะที แมนฯ ยูไนเต็ด กับ เชลซี ต่างพลาดทำแฮตทริกของตัวเองไปแล้ว
18 |     ในการปรับเรตล่าสุดของร้านพลซันน่าอย่าง สกาย เบท หลังจากเกม อาร์เซนอล
19 |     เบียดชนะ คริสตัล พาเลซ 1-0 เมื่อวันจันทร์
20 |     มีการปรับลดอัตราจ่ายแชมป์ของเต็ง 2 อาร์เซนอล ลงเล็กน้อย จาก 9/2 เป็น
21 |     4/1 อย่างไรก็ตาม เต็งแชมป์ยังคงเป็น แมนฯ ซิตี้ แชมป์ 3 สมัยซ้อน
22 |     ตามเดิม ในเรต 2/3ส่วนทาง เชลซี ถูกปรับเพิ่มอัตราจ่าย
23 |     ไปเป็นตัวเต็งลำดับที่ 5 ตามหลังทั้ง แมนฯ ยูไนเต็ด และ นิวคาสเซิล
24 |     ไปแล้ว หลังจากแพ้ว เวสต์แฮม เมื่อวันอาทิตย์
25 |   </p>
26 |
27 |   return <p> Mahanakorn News!!!!!!!!!!!! </p>;
28 | }
29 |
```

Error จำแดงเล็กน้อย เนื่องจากเราใช้ไม่ถูกวิธี ให้ปรับเปลี่ยนเป็นวิธีนี้ครับ กรณีที่เราต้องการ return หลายบรรทัดต้อง ให้อยู่ในวงเล็บ (สังเกตการ return ในไฟล์ App.tsx)

```

6 function MyComponent() {
7   return (
8     <p>
9     บริษัทรับพนันถูกกฎหมายของอังกฤษ มีการปรับเรตลดอัตราจ่ายการที่
10    อาร์เซนอล จะครองแชมป์ พรีเมียร์ลีก 2023/24 ลงเล็กน้อย
11    หลังจากทีมปืนใหญ่เป็นอีกรายที่เดินหน้าชนะ 2 เกมซ้อน ในขณะที่ เชลซี
12    ถูกเพิ่มเรตจนกลายเป็นเต็ง 5 ตามหลัง แมนเชสเตอร์ ยูไนเต็ด ไปแล้ว
13    พรีเมียร์ลีก ผ่านสัปดาห์ที่ 2 ไป โดยมี โบรห์ตัน, แมนเชสเตอร์ ซิตี้
14    และ อาร์เซนอล เป็นเพียง 3 ทีมที่กวาด 6 แต้มเต็มเข้ากระเป๋
15  </p>
16  <p>
17    ในขณะที่ แมนฯ ยูไนเต็ด กับ เชลซี ต่างพลาดท่าแพ้ครั้งแรกของตัวเองไปแล้ว
18    ในการปรับเรตล่าสุดของร้านพูลชั้นนำอย่าง สกาย เบท หลังจากเกม อาร์เซนอล
19    เบียดชนะ คริสตัล พาเลซ 1-0 เมื่อวันจันทร์
20    มีการปรับลดอัตราจ่ายแชมป์ของเต็ง 2 อาร์เซนอล ลงเล็กน้อย จาก 9/2 เป็น
21    4/1 อย่างไรก็ตาม เติ้งแชมป์ยังคงเป็น แมนฯ ซิตี้ แชมป์ 3 สมัยซ้อน
22    ตามเดิม ในเรต 2/3 ส่วนทาง เชลซี ถูกปรับเพิ่มอัตราจ่าย
23    ไปเป็นตัวเต็งลำดับที่ 5 ตามหลังทั้ง แมนฯ ยูไนเต็ด และ นิวคาสเซิล
24    ไปแล้ว หลังจากแพ เวสต์แฮม เมื่อวันอาทิตย์
25  </p>
26  );
27  |
28  }

```

พอเราใส่วงเล็บให้การ return อ้าววว โปรแกรมก็ยัง Error อยู่ดี เหตุผลก็เพราะว่าในการ return เราสามารถส่งกลับได้ทีละ Element เท่านั้น ซึ่งจากตัวอย่างเรามี <p></p> 2 ครั้ง ทำให้มันเกิด Error ขึ้นมา ซึ่งเราสามารถแก้ไขได้ด้วยการใส่ div

```

-
6 function MyComponent() {
7   return (
8     <div>
9     <p>
10    บริษัทรับพนันถูกกฎหมายของอังกฤษ มีการปรับเรตลดอัตราจ่ายการที่ อาร์เซนอล
11    จะครองแชมป์ พรีเมียร์ลีก 2023/24 ลงเล็กน้อย
12    หลังจากทีมปืนใหญ่เป็นอีกรายที่เดินหน้าชนะ 2 เกมซ้อน ในขณะที่ เชลซี
13    ถูกเพิ่มเรตจนกลายเป็นเต็ง 5 ตามหลัง แมนเชสเตอร์ ยูไนเต็ด ไปแล้ว
14    พรีเมียร์ลีก ผ่านสัปดาห์ที่ 2 ไป โดยมี โบรห์ตัน, แมนเชสเตอร์ ซิตี้
15    และ อาร์เซนอล เป็นเพียง 3 ทีมที่กวาด 6 แต้มเต็มเข้ากระเป๋
16  </p>
17  <p>
18    ในขณะที่ แมนฯ ยูไนเต็ด กับ เชลซี ต่างพลาดท่าแพ้ครั้งแรกของตัวเองไปแล้ว
19    ในการปรับเรตล่าสุดของร้านพูลชั้นนำอย่าง สกาย เบท หลังจากเกม อาร์เซนอล
20    เบียดชนะ คริสตัล พาเลซ 1-0 เมื่อวันจันทร์
21    มีการปรับลดอัตราจ่ายแชมป์ของเต็ง 2 อาร์เซนอล ลงเล็กน้อย จาก 9/2 เป็น
22    4/1 อย่างไรก็ตาม เติ้งแชมป์ยังคงเป็น แมนฯ ซิตี้ แชมป์ 3 สมัยซ้อน ตามเดิม
23    ในเรต 2/3 ส่วนทาง เชลซี ถูกปรับเพิ่มอัตราจ่าย ไปเป็นตัวเต็งลำดับที่ 5
24    ตามหลังทั้ง แมนฯ ยูไนเต็ด และ นิวคาสเซิล ไปแล้ว หลังจากแพ เวสต์แฮม
25    เมื่อวันอาทิตย์
26  </p>
27  </div>
28  );
29  }

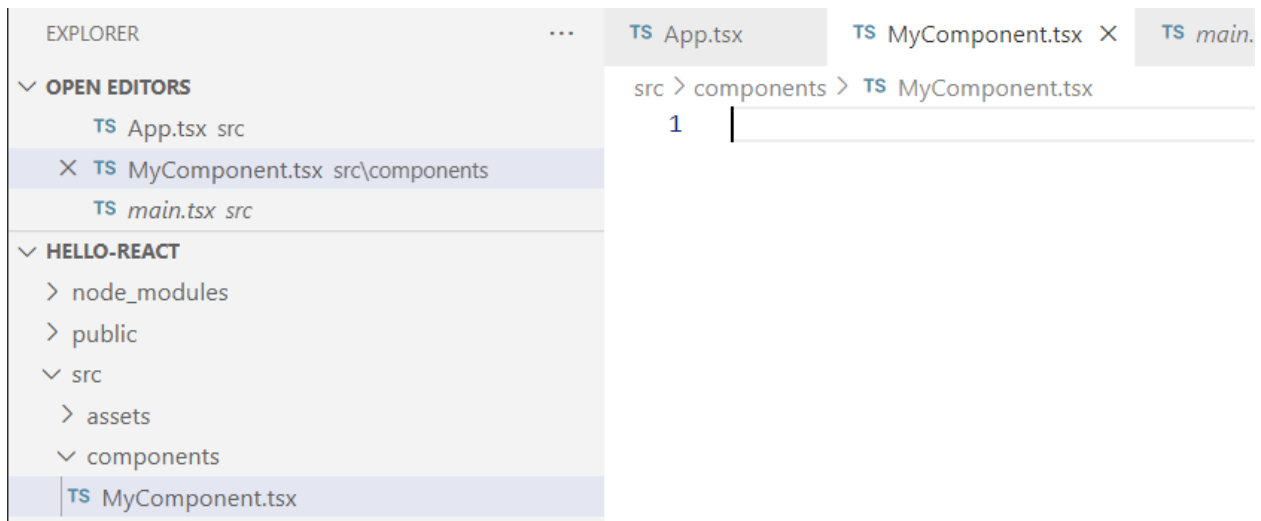
```

ที่นี้ก็สวยงามละครับ แต่ <div> ในไฟล์มันอาจจะไปชนกับ CSS เราได้นะ ดังนั้นเราสามารถแก้ไข <..</> ก็ได้

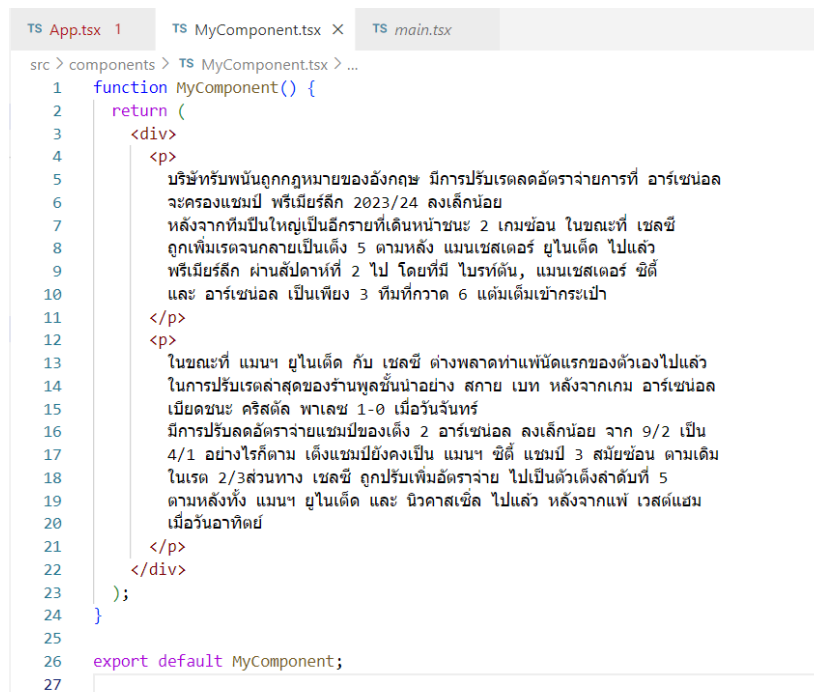
การแยก Component เป็นไฟล์

เวลาเราเขียนโปรแกรมแยกๆ กันส่วนใหญ่แล้วเราจะไม่เขียนในไฟล์ๆ เดียวกันหลายๆ คนหรอก ควรจะแบ่งแยกกันไปเขียนแล้วค่อยให้หน้าหลักเรียกไฟล์ๆ นั้นขึ้นมา ดังนั้น เราจะแยก MyComponent ไปเขียนเป็นอีก 1 ไฟล์

- สร้าง Folder components ขึ้นมา
- สร้างไฟล์ชื่อ MyComponent.tsx ใน Folder



ทำการ cut ข้อมูลทั้งหมดใน Function MyComponent มาจาก App.tsx แล้วพิมพ์บรรทัดที่ 26 เพิ่มมาด้วยนะครับ



ตอนนี้หน้าเว็บเราจะ Error นะครับคือมันไม่แสดงผล เพราะตอนที่เราริเรียก <MyComponent/> มันจะยังไม่เจอข้อมูลอะไรเลย เพราะ Function เราไปเขียนเป็นไฟล์ซะแล้ว ดังนั้นต้องทำการ Import มาก่อนตามรูปครับ

```
TS App.tsx  TS MyComponent.tsx  TS main.tsx
src > TS App.tsx > ...
1  import { useState } from "react";
2  import reactLogo from "../assets/react.svg";
3  import viteLogo from "/vite.svg";
4  import "../App.css";
5
6  import MyComponent from "../components/MyComponent";
7
```

VS code มันฉลาดเราพิมพ์นิดเดียว เดี๋ยวมันเพิ่มให้เองครับ ทีนี้โปรแกรมเราก็สามารถทำงานได้ตามปกติละ

3. JSX และ Component

- JSX เป็น Markup Syntax หน้าตามัน คล้ายๆกับ HTML
- เราสามารถแทรกโค้ด JavaScript ไปได้ หรือสั่ง execute javascript ใน JSX ได้
- ใน markup เวลาเราใช้ `class` เราจะใช้ `className` แทน
- ต้องมี close tag เสมอ
- **Single Root Element** ห้าม return multiple element (ต้อง wrap ให้เป็น 1 element)

ตัวอย่าง JSX ที่ผิด คือ ไม่สามารถ return multiple element แบบนี้ได้ และตัวอย่างที่ถูกต้อง คือ wrap เป็น 1 element

```
// ✗ return multiple element ไม่ได้
function MyComponent() {
  return (
    <h1>Hello World!</h1>
    <p>This is component!</p>
  )
}

// ✔ wrap มันด้วย div ซะ
function MyComponent() {
  return (
    <div>
      <h1>Hello World!</h1>
      <p>This is component!</p>
    </div>
  )
}
```

ในบางครั้ง เราไม่ต้องการ wrap มันด้วย `div` เพิ่ม เพราะมันจะทำให้ เวลาเรา render HTML เรามีโค้ด `div` เพิ่มมา ทำให้ stylesheet ของเราผิดเพี้ยน เราสามารถ wrap element ด้วย การใช้ `React.Fragment` หรือ ตัวย่อ คือ `<>` และ `</>` แบบนี้

```
function MyComponent() {
  return (
    <React.Fragment>
      <h1>Hello World!</h1>
      <p>This is component!</p>
    </React.Fragment>
  )
}

// หรือใช้ short hand
function MyComponent() {
  return (
    <>
      <h1>Hello World!</h1>
      <p>This is component!</p>
    </>
  )
}
```

การใส่ค่า JavaScript หรือการแทรกโค้ด JavaScript ลงไปใน JSX เราจะใช้ `{ }` ตัวอย่าง

```
function MyComponent() {
  const title = 'Hello world!'
  const message = 'This is component!'

  return (
    <div>
      <h1>{title}</h1>
      <p>{message}</p>
    </div>
  )
}
```

5. Rendering List

กรณีที่เรต้องการ render Component ที่คล้ายๆ กันเช่น Profile หรือ list ของ สิ่งของต่างๆ เราสามารถใช้ JavaScript Array map ได้ ตัวอย่าง ที่เห็นบ่อยๆ เช่น เราจะแสดง และข้างในมี มีข้อมูลดังนี้

```
<ul>
  <li>John Doe</li>
  <li>Chuck Norris</li>
  <li>Jane Doe</li>
  <li>Foo Bar</li>
  <li>Alice Bob</li>
</ul>
```

สิ่งที่ต้องทำคือเปลี่ยนข้อมูล ให้อยู่ในรูปแบบ JavaScript Array

```
const names = [
  'John Doe',
  'Chuck Norris',
  'Jane Doe',
  'Foo Bar',
  'Alice Bob'
]
```

2. ใช้ map เพื่อแปลงให้เป็น array ของ JSX nodes ชื่อ users

```
const users = names.map(name => <li>{name}</li>)
```

3. จากนั้น ใน component เราก็สามารถ ใช้แบบนี้ได้

```
function MyComponent() {
  return <ul>{users}</ul>
}
```

นอกจากนี้ เราสามารถแทรก Map ใน JSX ตรงๆ ภายใน return function ได้เลย โดยไม่ต้องสร้างตัวแปรก็ได้ แบบนี้

```
function MyComponent() {
  return (
    <ul>
      {
        users.map(user => {
          return <li>{user}</li>
        })
      }
    </ul>
  )
}
```


ทดลองการทำงาน โดยเพิ่มคำสั่งตามรูป

```
TS App.tsx TS MyComponent.tsx X TS main.tsx
src > components > TS MyComponent.tsx > MyComponent
1  function MyComponent() {
2      const names = [
3          "John Doe",
4          "Chuck Norris",
5          "Jane Doe",
6          "Foo Bar",
7          "Alice Bob",
8      ];
9      const users = names.map((name) => <li>{name}</li>);
10
11     return (
12         <>
13             <p>
14                 บริษัทรบพันถูกกฎหมายของอังกฤษ มีการปรับเรตลดอัตราจ่ายการที่ อาร์เซนอล
15                 จะครองแชมป์ พรีเมียร์ลีก 2023/24 ลงเล็กน้อย
16                 หลังจากทีมปืนใหญ่เป็นอริราชที่เด่นหน้าชนะ 2 เกมซ้อน ในขณะที่ เชลซี
17                 ถูกเพิ่มเรตจนกลายเป็นเต็ง 5 ตามหลัง แมนเชสเตอร์ ยูไนเต็ด ไปแล้ว
18                 พรีเมียร์ลีก ผ่านสัปดาห์ที่ 2 ไป โดยที่มี โบรห์ตัน, แมนเชสเตอร์ ซิตี้
19                 และ อาร์เซนอล เป็นเพียง 3 ทีมที่กวาด 6 แต้มเต็มเข้ากระเป๋
20             </p>
21             <p>
22                 ในขณะที่ แมนฯ ยูไนเต็ด กับ เชลซี ต่างพลาดทำแฮตทริกของตัวเองไปแล้ว
23                 ในการปรับเรตล่าสุดของร้านพูลชั้นนำอย่าง สกาย เบท หลังจากเกม อาร์เซนอล
24                 เบียดชนะ คริสตัล พาเลซ 1-0 เมื่อวันจันทร์
25                 มีการปรับลดอัตราจ่ายแชมป์ของเต็ง 2 อาร์เซนอล ลงเล็กน้อย จาก 9/2 เป็น
26                 4/1 อย่างไรก็ตาม เต็งแชมป์ยังคงเป็น แมนฯ ซิตี้ แชมป์ 3 สมัยซ้อน ตามเดิม
27                 ในเรต 2/3 ส่วนทาง เชลซี ถูกปรับเพิ่มอัตราจ่าย ไปเป็นตัวเต็งลำดับที่ 5
28                 ตามหลังทั้ง แมนฯ ยูไนเต็ด และ นิวคาสเซิล ไปแล้ว หลังจากแพ้ว เวสต์แฮม
29                 เมื่อวันอาทิตย์
30             </p>
31             <p>{users}</p>
32         </>
33     );
34 }
35
36 export default MyComponent;
37
```

Condition Rendering If-else

ที่ components ให้สร้างไฟล์ใหม่ชื่อ Myifelse.tsx แล้วเขียนโปรแกรมตามรูป

```
TS App.tsx    TS MyComponent.tsx    TS Myifelse.tsx X
src > components > TS Myifelse.tsx > ...
1  function Myifelse() {
2    let result;
3    const score = 50;
4
5    if (score >= 50) {
6      result = <p>You've passed!</p>;
7    } else {
8      result = <p>Please try again next time</p>;
9    }
10
11   return (
12     <div>
13       <p>Your result:</p>
14       {result}
15     </div>
16   );
17 }
18
19 export default Myifelse;
```

ทดลอง import แล้วเรียกใช้ Fuction myifelse ดูครับ

คำสั่งที่เหลือ

Events

Props

State

สร้างโปรแกรมรับคะแนนแล้วตัดเกรด