

รายงาน LAB 3SA03

เรื่อง 3SA03 - React

จัดทำโดย

นายอุสมาน สุหลง

รหัสนักศึกษา 5935512036

Sec02

เสนอ

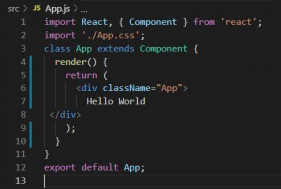
Teacher: THITINAN KLIANGSUWAN

ภาคเรียนที่1 ปีการศึกษา2562

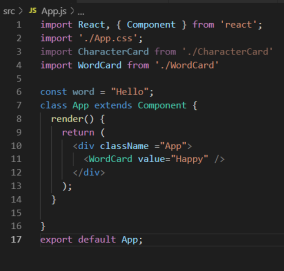
สาขาวิศวกรรมคอมพิวเตอร์

คณะวิศวกรรมศาสตร์

***Check point***

แสดง Hello World

ผลลัพธ์



1. App.js ในส่วนของไฟล์ App.js เป็นไฟล์ที่หน้าที่แสดงหน้าต่างผลของการรันโปรแกรม ซึ่งในไฟล์ App.js นี้มีการเรียกใช้ไฟล์ อื่นภายในตัวด้วย เช่น การเรียกใช้ไฟล์ App.css , การเรียกใช้ไฟล์ WorldCard.js



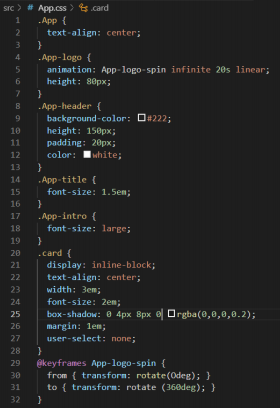
2. CharacterCard.js CharacterCard.js เป็นการท าให้ค่า values ที่เราก าหนดมานั้น สามารถกดได้ ซึ่งไฟล์ CharacterCard.js มีการ import จากไฟล์ App.css ในเรื่องของรูปแบบการจัดและสีของตัวอักษร และในไฟล์CharacterCard.js มีการประกาศฟังก์ชันต่าง ๆ เช่น ฟังก์ชัน activate , componentDidUpdate และมีการประกาศ constructor เป็น state มี active : false การประกาศเป็น state เสมือนกับว่า state เป็นตัวแม่ ฟังก์ชันอื่นๆเป็นลูก ๆ ดังนั้นเวลาฟังก์ชันอื่นต้องการใช้ค่าก็ สามารถดึงค่านั้นไปใช้ได้ผ่าน ผ่านตัว this.state.active

3. WordCard.js ในส่วนของไฟล์ Wordcard.js เป็นไฟล์ที่ใช้สำหรับการสบับต แหน่งของค่า values ที่เรากำหนดจากไฟล์ App.js และ ภายในไฟล์มีการประกาศใช้ฟังก์ชัน componentWillMount , activationHandler , prepareStateFromWord



4. App.css เป็นไฟล์สำหรับกำหนด layout ต่าง ๆ ที่ใช้ ที่ไฟล์อื่น หรือ components อื่นสามารถนำไปใช้ได้

Code การทำงาน



ผลการทดลอง



Link : <https://github.com/5935512036/2SA03>

***งานท้ายการทดลอง***

หน้าเว็บเกม



โจทย์เพิ่มเติม

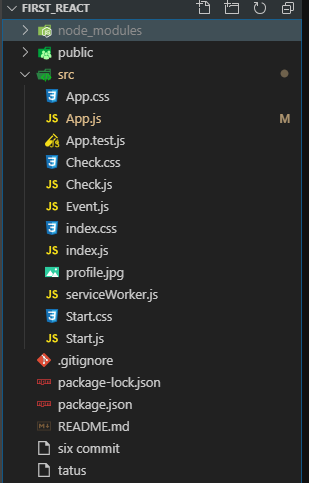
1. ให้ซ่อน Complete เมื่อตอบผิด
2. ทำให้ไม่สามารถเลือกปุ่มเดิมซ้ำกันได้
3. ให้รีเซ็ทหน้าเมื่อตอบเสร็จ (โค๊ดอยู่ในแลบชีท)

สิ่งที่คิดเพิ่มจากตัวอย่างในLAB



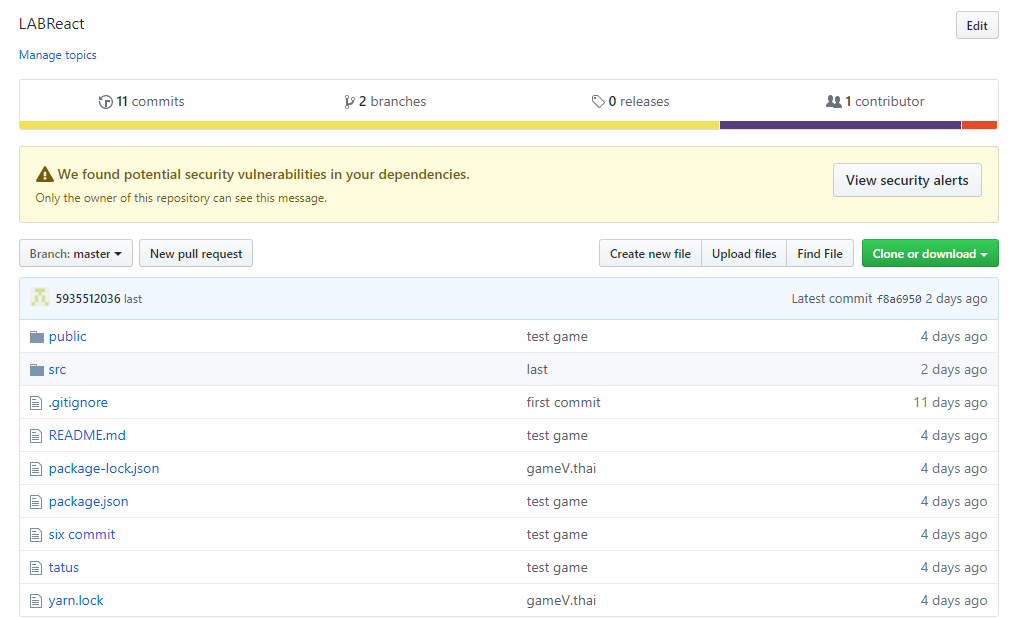
กำหนดให้เล่น 3 ครั้ง ถ้าหากตอบได้ก่อนให้มีปุ่มเล่นอีกครั้งพร้อมบอกว่าชนะ

ไฟล์ทั้งหมดในfolder

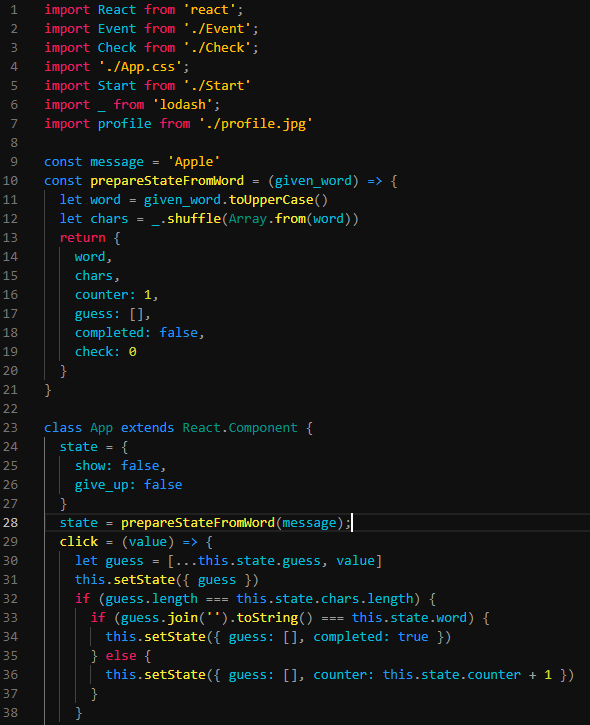


อัพขึ้น gibhub 11 commit

Link : <https://github.com/5935512036/2SA03>



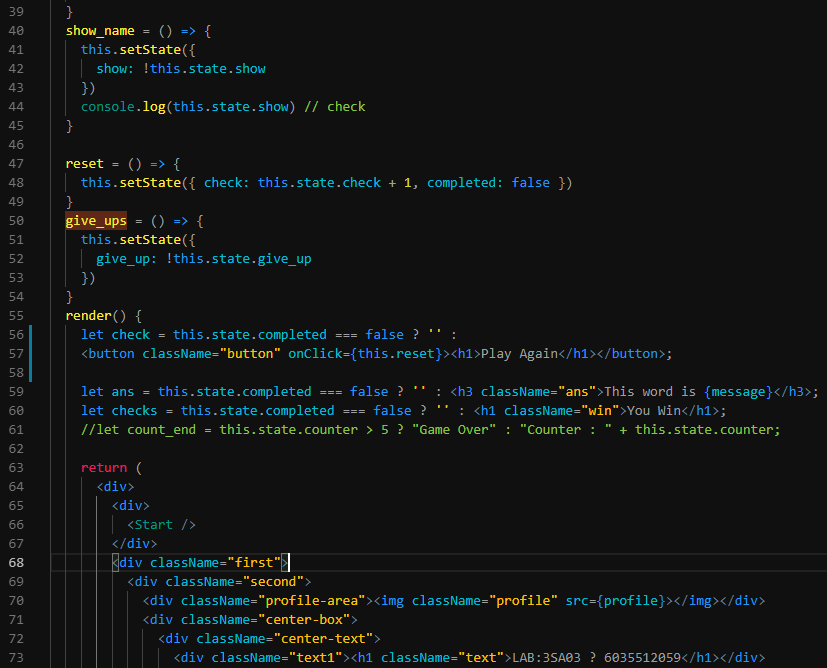
**Code App.js**



1. Import ไฟล์ต่างๆเพือที่จะสามารถใช้codeหรือคำสั่งจากไฟล์ที่Import
2. เป็นฟังก์ชันที่returnค่าstateมา

Stateสำหรับจัดการข้อความโดยผมจะเพิ่มตัวแปรcheckเพื่อที่จะให้มันresetใหม่(เริ่มใหม่)

1. เป็นการต่อตัวอักษรให้เป็นข้อความ แล้วไปเปรียบเทียบกับคำmassageที่แปลงเป็นตัวใหญ่แล้ว
2. ถ้ำคำที่ทายมาไม่ถูกให้ทำการเดาเรื่อยๆแล้วเพิ่มค่าcounterไปหนึ่ง
3. ผมจะเพิ่มตัวstateที่โชว์คำใบ้ กับ ชื่อของผม

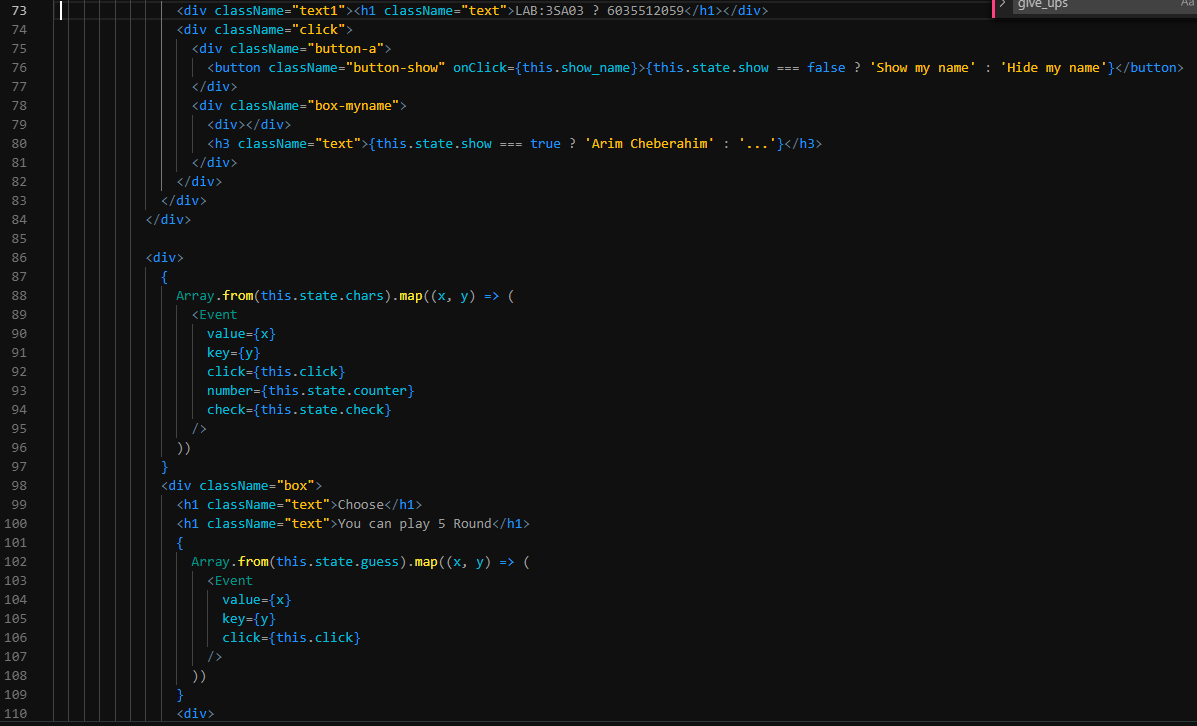


1.เปลียนค่าstateให้ตรงกันข้ามเรื่อยๆ หากมีการกด

2.กำหนดตัวแปรรับค่าที่ตรงเงื่อนไขเป็นปุ่มเริ่มเกมใหม่

3.ตัวแสดงรูปภาพ

4.เรียกใช้งานไฟล์Start.js



5.เป็นการแยกตัวอักษรแล้วส่งไปให้ Evant.jsจัดการต่อส่งตัวเมคทอดclickตัวแปรcounter,checkที่เก็บค่าstateแต่ล่ะตัว

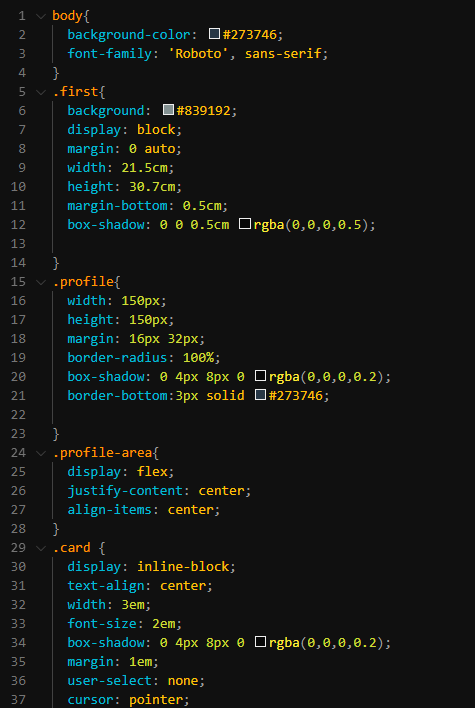
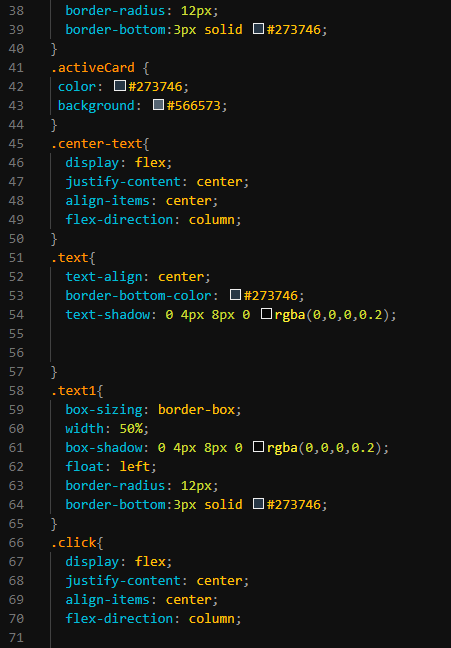


6.แสดงค่าที่ตัวแปร check,ans,checks เก็บไว้ดูได้จากโค้ดบรรทัด56-60

**Code App.css**

1.กำหนดสีbackgrond กับ font ในbody

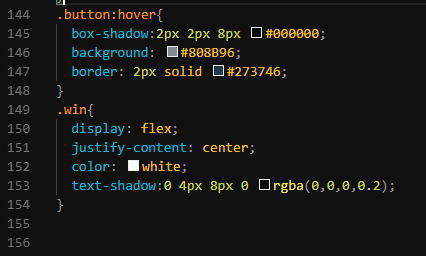
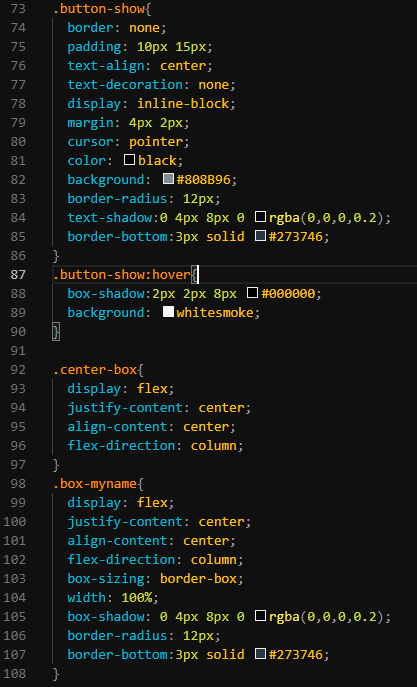
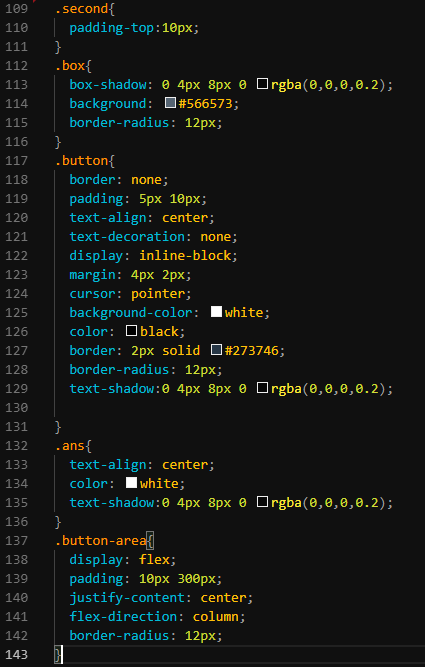
Bodyเป็นclass ใหญ่สุดของbackground



4.Class text กำหนดให้ข้อความของเราอยู่ตรงกลาง แล้วกำหนดสีของเส้นขอบล้างกับเงาของข้อความด้วย

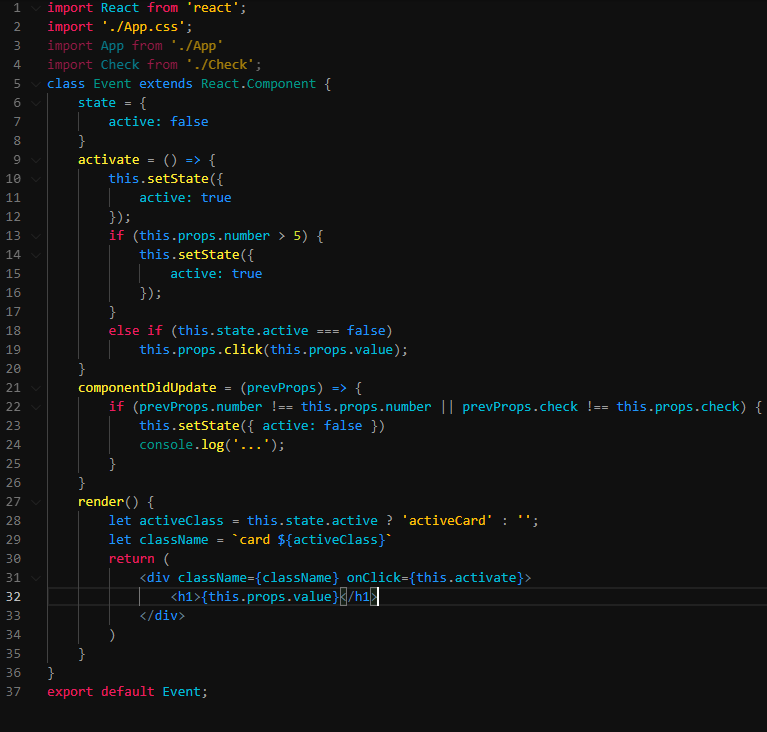
2.เป็นสวนของรูปprofileโดยกำหนดความกว้างความสูง ระยะห่าง องศาของขอบรูป เงา เส้นขอบใต้ภาพ

3.กำหนดให้รูปอยู่ตรงกลาง



5.คำสั่ง border เป็นการเพิ่มกรอบของสิ่งนั้นๆ solid เป็นเส้นทึบทำงานคล้ายๆsrokeในphotoshop

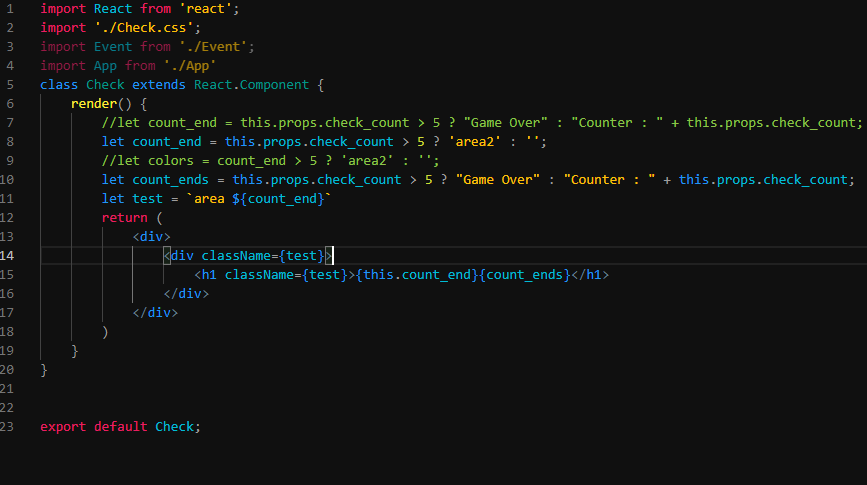
Code Event.js (เป็นไฟล์ที่เอาตัวแปร เมคทอด state มาใช้งานแล้วส่งค่ากลับ)



1.เป็นเงื่อนไขที่เช็คว่าถ้าตัวอักษรที่เราคลิ้กมันยังไม่ถูกactiveหรือยังไม่มีการกดให้เข้าเงื่อนไขนี้ได้(ทำให้ไม่สามารถกดซ้ำได้)

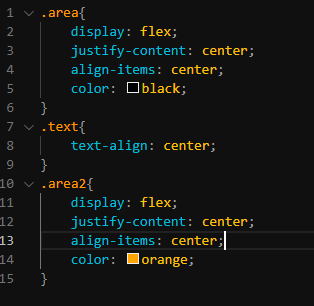
2.เป็นส่วนของการอัพเดทคลาสมาแสดงโดยเราจะเก็บคลาสที่จะอัพเดทไว้ในตัวแปรต้องมีตัว$ด้วย

**Code Check.js**



1.ชื่อclassที่ทำตัวอัพเดทไว้โดยclass area จะแสดงอยู่แล้วแต่ถ้าหากเข้าเงื่อนไขที่ตัวแปรtestเก็บไว้มันก็จะอัพเดทตัว css ทันที

Code Check.css



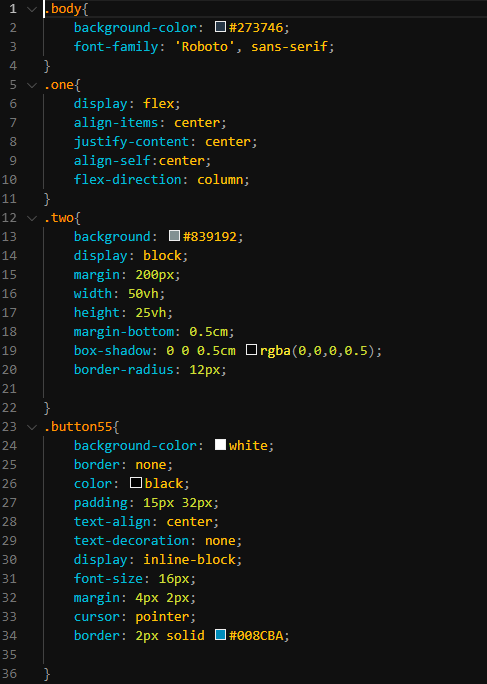
**Code Start.js**



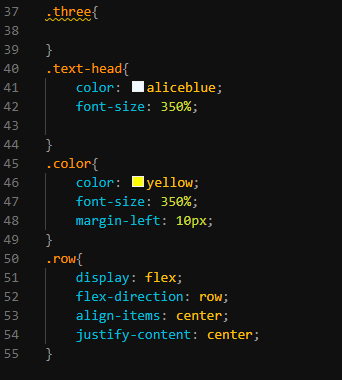
2.มันคือตัวที่link ไปหน้า .js หรือ .html อื่น

1.เป็นการImport routerแต่ผมไม่ได้ใช้ในการเขียนครั้งนี้เพราะมันจะทำให้ตัวจัดไฟล์Css ต้องจัดใหม่หมดเพราะผมมาทำไอนี้ทีหลังผมเลยตั้งทิ้งไว้เฉยๆแต่รู้วิธีการใช้แล้ว

**Code Start.css**



3.คลาสone ของผมเป็นคลาสที่จัดทุดอย่างที่อยู่ในนี้ให้อยู่ตรงกลางแล้วเรียนเป็นแถวๆตาม Div



4.Class color กำนดสีของข้อความ แล้วขนาดของข้อความกับระยะห่างทางด้านซ้าย