

1. Explain the what's similar & difference

between cookie / localStorage / sessionStorage.

Cookie localStorage sessionStorage นั้นเหมือนกันที่ไว้ใช้ในการเก็บข้อมูลโดยที่ Cookie จะเป็นการเก็บข้อมูลเพื่อส่งไปยัง Server ซึ่งสามารถหมดอายุได้โดยที่นำมาเพื่ออ่านจากทางด้านserverเป็นหลัก

LocalStorage เป็นการเก็บข้อมูลไว้ที่คอมพิวเตอร์โดยที่ไม่มีวันหมดอายุโดยที่เก็บเป็น key กับ value สามารถอ่านจากฝั่งclientเท่านั้น

SessionStorage เป็นการเก็บข้อมูลบนbrowserหรือ tab โดยที่ไม่ส่งไปยังServer โดยเมื่อปิดtab หรือ browserนั้นจะเป็นการลบ SessionStorage ทิ้งไป โดยเก็บข้อมูลเป็นkey valueเหมือน LocalStorage

2. Today React have hooks. Do we still need a class component? If your answer is yes then which case that we still need to use class component.

ยังต้องการอยู่เนื่องจากบาง lifecycle ของ react ยังจำเป็นที่จะต้องใช้ class component อยู่เช่น componentDidMount

3. Briefly describe how *Virtual DOM* works.

การทำงานของVirtual DOMคือเมื่อมีการเปลี่ยนแปลงเกิดขึ้น Virtual DOMจะทำการ re-render หน้า UI ทั้งหมดใน Virtual DOM เพื่อนำมาเปรียบเทียบกับ Virtual DOM ก่อนหน้านี้ว่ามีอะไรเปลี่ยนแปลงไปบ้าง real DOM จะทำการเปลี่ยนแปลงตามที่ Virtual DOM มีการเปลี่ยนแปลงมา

4. Consider this React's components tree

```
Apps > ComponentA > ComponentB > ComponentC >  
ComponentD > ComponentE
```

If we have a state at Apps component, and ComponentE component want to access that state value. How do **you implements** this?

ใช้การpropsเข้าไปในcomponent A ,B ตามลำดับไปเรื่อยๆ

5. What different between using relative / absolute / fixed to position the element.

Position relative คือ การที่เราสามารถใส่ค่าบนล่างซ้ายขวาเพื่อให้มันถูกปรับจากตำแหน่งปกติโดยที่เนื้อหาอื่นยังเหมือนเดิม

Position absolute เป็น position แสดงผลเป็นอิสระโดยที่เรากำหนดให้ไปอยู่ตรงไหนก็ได้และถูกเลื่อนไปเมื่อเราscroll mouse

Position fixed เป็น position ที่เกี่ยวข้องกับ viewport โดยที่เมื่อถูกวางตรงไหนแล้วมันจะอยู่ที่เดิมถึงแม้ว่าเราจะscroll mouseก็ตาม

6. Explain the different between using callback / Promise / async await. When to use and when not to.

Callback คือการที่เราทำfunction หนึ่งเสร็จแล้วให้อีกfunction หนึ่งทำงานต่อซึ่งจะทำงานต่อกันเป็นทอดๆเป็นการทำงานที่ทำให้โค้ดนั้นดูยุ่งยากและอาจจะเกิดปัญหาcallback hell ได้ จึงไม่แนะนำให้ใช้

Promise เป็น object พิเศษที่เก็บค่าบางอย่างสามารถเข้าถึงได้โดยใช้ .then ซึ่งการทำงานเหมือนcall back แต่ว่าทำให้โค้ดอ่านได้เข้าใจง่ายขึ้นและทำให้การดักจับerrorทำได้ง่ายขึ้นโดยใช้ .catch ไม่ควรใช้ถ้าต้องมีการทำงานต่อกันมหลายขั้นตอน

Async await ถ้าเราประกาศ async ไว้หน้า function ตัว function นั้นจะกลายเป็น synchronous ทันทีโดยที่ await มาช่วยให้รอ promise ของเราทำงาน

เสร็จก่อนและสามารถhandle error ด้วย try/catch ได้การเขียนด้วยวิธีนี้จะทำให้
โค้ด คลีนและเขียนได้ง่ายขึ้นมาก