1. reference

<https://www.30secondsofcode.org/react/s/use-interval>

1. 구현

const useInterval = (callback, delay) => {

   const savedCallback = useRef();

   useEffect(() => {

      savedCallback.current = callback;

   }, [callback]);

   useEffect(() => {

      function tick() {

         savedCallback.current();

      }

      if (delay !== null) {

         let id = setInterval(tick, delay);

         return () => clearInterval(id);

      }

   }, [delay]);

};

1. 사용

useInterval(() => {

      setSeconds((prev) => prev + 1);

      setCurrentPhoto(seconds % 4);

   }, 10000);

1. 원리

setInterval 함수 : 수행할 작업과, miliseconds 시간을 받아 일정시간마다 그 함수를 실행하도록 함.

1. useInterval은 수행할 작업(callback)과 시간(delay)을 받음. 그리고 안에서 callback을 저장하는데, callback이 변경될때마다 다시 저장하여야하므로, useEffect안에서 저장함.
2. 이후, 그 저장된 callback을 다시 함수 안에서 실행시켜줌.
3. 그리고 delay가 있으면 setInterval로 일정시간마다 실행시키도록 함. 그리고 컴포넌트가 분리될 시 setInterval이 멈추도록 return문에 적어줌.
4. 또 delay가 변경될 때 다시 설정해줘야하므로 depedencies에 delay를 넣어줌.