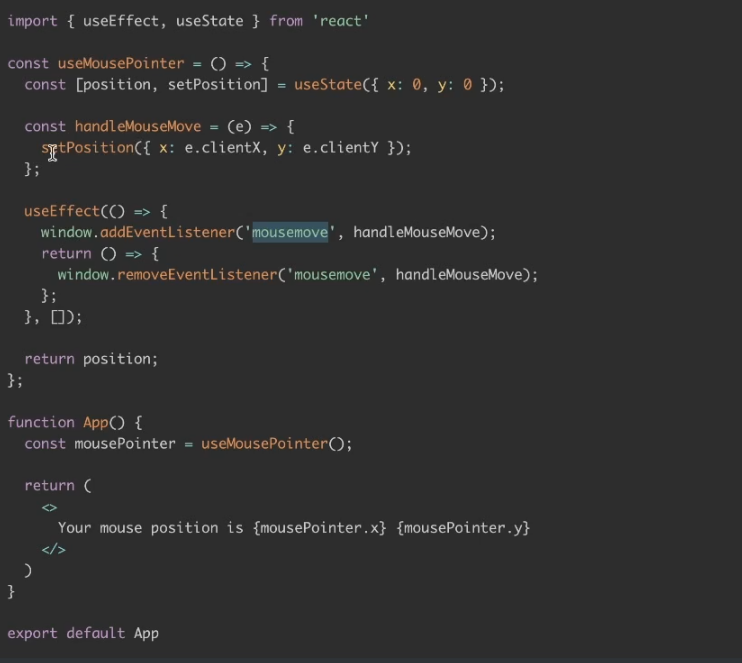
**useMousePointer Hook**



**useInterval**

import React, { useState } from "react";

import { useEffect, useRef } from "react";

// This is the custom hook

function useInterval(callback, delay) {

const savedCallback = useRef();

// Keep the latest version of the callback

useEffect(() => {

savedCallback.current = callback;

}, [callback]);

// Set up the interval

useEffect(() => {

if (delay === null) return;

const id = setInterval(() => {

savedCallback.current(); // call the latest callback

}, delay);

return () => clearInterval(id); // cleanup

}, [delay]);

}

function App() {

const [count, setCount] = useState(0);

useInterval(() => {

setCount(prev => prev + 1);

}, 1000); // every 1 second

return (

<div>

<h2>Counter: {count}</h2>

</div>

);

}

export default App;