React: hooks

useState y useEffect

A continuación veremos como aplicar useState y useEffect en React agregando eventos y funciones a los nodos correspondientes, usando react developer tools y creando interacción en la web.

import React, { useState, useEffect } from "react";

import ReactDOM from "react-dom/client";

const root = ReactDOM.createRoot(document.getElementById("root"));

function Counter() {

const [counter, setCounter] = useState(10);

useEffect(()=>{

console.log("render")

},[]);/\*esto indicará que el componente se renderizó correctamente 1 sola vez\*/

return (

<div>

<h1>Counter:{counter}</h1>

<button

onClick={() => {

setCounter(counter + 1);

}}

>

Sumar

</button>

<button

onClick={() => {

setCounter(counter - 1);

}}

>

Restar

</button>

<button

onClick={() => {

setCounter(0);

}}

>

Reiniciar

</button>

<hr/>

</div>

);

}

function Mensaje() {

const [mensaje, setMensaje] = useState("");

useEffect(()=>{

console.log("render")

},[mensaje]);/\*esto indicará que el componente se renderizó correctamente, el evento se ejecutará cada vez que se escriba y no por unica vez\*/

return (

<div>

<input

/\*mostraremos en tiempo real lo que escribe el usuario en consola\*/

/\*onChange={e=>console.log(e.target.value)}

/\*mostramos el mensaje en componentes con react dev tools\*/

onChange={e=>setMensaje(e.target.value)}

></input>

<button onClick={()=>{

alert("el mensaje es: "+ mensaje)

}}>Guardar</button>

</div>

);

}

root.render(

<>

<Counter />

<Mensaje />

</>

);