

Bootcamp de Full Stack F2

Bienvenidos a la clase N°44

- Hooks
 - useRef
 - useEffect
- Ejercicio práctico con useState

React

Hooks

Reglas de los Hooks

- **Los Hooks solo deben ser llamados en el nivel superior**
No invocar un o más Hook dentro de bucles, condiciones o funciones anidadas. De esta manera, se preserva la correctitud del estado de los Hooks porque se garantiza que los mismos siempre se ejecuten en el mismo orden cada vez que un componente se renderiza.
- **Los Hooks solo deben llamarse en funciones de React**
No invocar un o más Hooks desde funciones regulares de JavaScript. En vez de eso, se puede:
 - Invocar desde componentes funcionales de React o;
 - Invocar desde Hooks personalizados.
- En el modo estricto de React, se realizan dos llamados a las funciones de un componente con el objetivo de ayudar a encontrar impurezas accidentales. Este es un comportamiento sólo de desarrollo y no afecta en producción.

React

Hooks

Hook useRef

Este hook devuelve un objeto de referencia mutable cuya propiedad es **.current** y se inicializa por medio de un argumento. El objeto devuelto se mantendrá persistente durante la vida completa del componente.

```
const nombreDeVariable = useRef(initialValue);  
console.log(nombreDeVariable.current);
```

Consideraciones:

Cuando se cambia la propiedad **ref.current**, no se vuelve a renderizar el componente.
React no está al tanto de cuándo cambia el estado de una ref.

React

Hooks

Hook useEffect

Este hook nos permite agregar efectos secundarios a un componente funcional dado, es decir, permite realizar modificaciones en la lógica una vez se haya realizado el renderizado. El mismo, recibe dos parámetros, el primero es un callback y el segundo es un array de dependencias. Usar un array vacío de dependencias, significa que el efecto se ejecutará sólo una vez (al montar el componente).

```
useEffect( () => {  
    Bloque de código;  
}, [Dependencias]);
```

Una de las grandes ventajas de este hook es que simplifica el ciclo de vida, de forma que los tres métodos de clase (componentDidMount, componentDidUpdate y componentWillUnmount) pueden ser expresados utilizando únicamente este hook.

BREAK

Descansemos 15 minutos





JavaScript

Ejercicio Práctico con useState

Consigna:

- Crear una aplicación tipo CRUD utilizando useState. Se debe mostrar una lista de elementos en donde cada uno muestre el dato y dos botones (modificar y eliminar). Además, se debe mostrar un formulario para agregar o modificar un elemento.

Desafío Semanal N°13

Pair Programming



- Esta actividad se resuelve **fuera de clase** entre **dos estudiantes** y es de **carácter obligatorio**.
- La misma, se **aprueba** con **5 o más** y es calificada por nuestros tutores.
- En el caso de obtener una calificación **menor a 5**, se debe **re-entregar** dicha actividad.
- La fecha límite para **entregar** es el Miércoles **10/01/2024**.
- El siguiente link, contiene la consigna y los recursos necesarios para su desarrollo:
<https://drive.google.com/drive/folders/1MgwFAiHy8g76jmPgLVS7hRu3vzDzGRwE?usp=sharing>
- La entrega se realiza de forma individual en formato zip y se sube en la carpeta personal de Google Drive.
- Por último y no menos importante, registrar la entrega en el form que recibieron por mail.

CIERRE DE CLASE

Continuaremos en la próxima clase

