**Hooks React**

|  |  |  |
| --- | --- | --- |
| Hook | Propósito | Ejemplo común |
| **useState** | Manejar el estado. | Contador, manejar formularios. |
| **useEffect** | Manejar efectos secundarios (fetch, suscripciones). | Llamadas a APIs, actualizar el DOM. |
| **useRef** | Referencias mutables, acceso al DOM. | Enfocar inputs, almacenar valores persistentes entre renders. |
| **useMemo** | Memorizar valores calculados. | Cálculos costosos que dependen de ciertas variables. |
| **useCallback** | Memorizar funciones. | Optimización de funciones pasadas como props a componentes hijos. |

**1. useState**

El hook useState se utiliza para manejar el estado en un componente funcional.

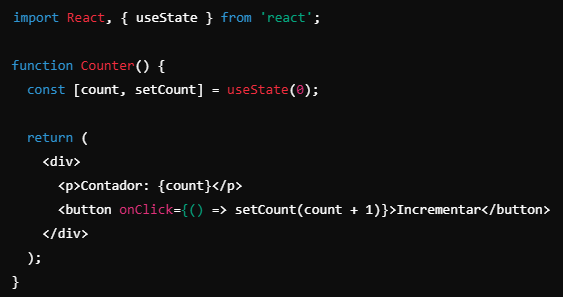
* **state**: Es la variable que contiene el estado actual.
* **setState: Es la función para actualizar el estado.**
* **initialValue: Es el valor inicial del estado.**

Sintaxis:

Interfaz de usuario gráfica, Texto, Sitio web

Descripción generada automáticamente

Ejemplo:



**2. useEffect**

El hook useEffect permite manejar efectos secundarios, como la interacción con APIs, suscripciones, y manipulación del DOM. Funciona de manera similar a los métodos del ciclo de vida como componentDidMount, componentDidUpdate y componentWillUnmount.

Texto

Descripción generada automáticamente

* **Callback**: Función que contiene el efecto.
* **Dependencies**: Lista de valores que controlan cuándo se ejecuta el efecto.

Ejemplo:

Texto

Descripción generada automáticamente