



wawiwa

React

2022 © Wawiwa Tech | Confidential



# 7 - States (estados)

2022 © Wawiwa Tech | Confidential



## ¿Qué es un State (estado)?

State es usado para **contener datos o información que pertenecen al componente**.

Estado del componente puede cambiar con el tiempo, cada vez que cambia el componente se re-renderiza (“se actualiza”).

Puedes tener más de una variable de estado



## useState

useState es un **React Hook** que te permite agregar una variable de estado a tus componentes. useState is usado en componentes funcionales.

### React Hooks:

Los Hooks son una nueva adición de React 16.8. Te permiten usar el estado y otras funcionalidades sin tener que escribir una clase.

### The syntax of useState:

```
const [state, setState] = useState(initialState);
```





## Sintaxis de useState

```
const [state, setState] = useState(initialState);
```

```
const [age, setAge] = useState(21);
```

**state** - el nombre del estado

**setState** - La función configurada retornada por useState te permite **actualizar el estado** a un valor diferente y dispara una re-renderización.



## Sintaxis de useState

```
const [state, setState] = useState(initialState);
```

```
const [age, setAge] = useState(21);
```

**useState** - useState es un Hook, entonces solamente lo puede llamarlo en el nivel superior de tu componente o en tus propios Hooks. No puedes llamarlo dentro de ciclos o condiciones. Si necesitas eso, extrae un nuevo componente y mueve el estado dentro de el.

**useState** retorna un arreglo con exactamente dos valores:

1. **El estado actual** - durante la primera renderización, va a coincidir con el initialState que pasaste.
2. **La función "set"** - te permite actualizar el estado a otro valor y dispara una re-renderización.



## Sintaxis de useState

```
const [state, setState] = useState(initialState);
```

```
const [age, setAge] = useState(21);
```

**initialState** - El valor que tu quieras que sea el inicial. Puede ser un valor de cualquier tipo o una función que lo calcule (pero hay un comportamiento especial para las funciones.

For example: `const [num, setNum] = useState(functionName);`

Si pasas una función como `initialState`, va a ser tratada como una *función inicializadora*. Debe ser pura, no debe tomar argumentos y debe retornar un valor de cualquier tipo. React va a llamar a tu función inicializadora cuando se inicialice el componente, y va a guardar el valor retornado como valor inicial.



## useState

Para usar este método, primero impórtalo:

```
import { useState } from "react";
```

Llama useState en el nivel superior de tu componente para declarar un variable de estado.

Ejemplos:

```
function App() {  
  const [age, setAge] = useState(21);  
  const [name, setName] = useState('Taylor');  
  
  return ( ...)
```





## ¿Cuándo usamos Estados?

Usa un estado cuando quieras actualizar/afectar lo que se muestra en la pantalla usando una variable.

Actualizar una variable en la pantalla no va a renderizar o cambiar nada.

Por ejemplo, puedes tener un componente contador que muestra una cuenta que cambia cada vez que se hace click a un botón. En este caso, necesitas guardar el valor del contador en un estado, para que el componente puede re-renderizar y mostrar el valor actualizado cuando el botón ha sido presionado.

En el siguiente capítulo, vamos a aprender cómo actualizar un valor cuando se hace click a un botón usando eventos de React.



## useState - Ejemplo simple sin efectos secundarios

```
import React, {useState} from 'react';

export default function App() {
  const [age, setAge] = useState(21);
  const [name, setName] = useState('Taylor');

  return (
    <>
      <div>Hello my name is {name} and I'm {age} years old!</div>
    </>
  );
}
```



## Ejercicio 4

Crea un componente Counter que muestre un valor de una cuenta y permita que el usuario incremente o reduzca la cuenta usando botones.

1. Crea un componente de React llamado Counter.
2. Dentro del componente Counter, importa el hook useState.
3. Declara una variable de estado llamada "count" usando el hook useState. Inicialízalo a 0.
4. Renderiza el valor actual de la cuenta dentro de un elemento <p>.
5. Renderiza dos botones: uno para incrementar la cuenta y otro para reducirla.

Count: 0

