**React**

**Desestructuración de arreglos.**

Documentación: <https://es.reactjs.org/>

<https://es.reactjs.org/docs/getting-started.html>

<https://developer.mozilla.org/es/docs/Web/JavaScript/Reference/Operators/Destructuring_assignment#desestructuraci%C3%B3n_de_arreglos>

La desestructuración de arreglos es mucho más fácil que la desestructuración de objetos y funcionan casi igual, solamente hay una diferencia entre ellos y es que en **la desestructuración de los arreglos, la posición es muy importante,** eso es toda la diferencia.

Para la desestructuración de un arreglo se usan los corchetes **“[]”** a diferencia de los objetos que se usan las llaves **“{}”.** Ejemplo:

const personajes = ['Goku', 'Vegeta', 'Trunks'];

const [Goku, Vegeta, Trunks] = personajes;

console.log( Goku, Vegeta, Trunks );

Pero por ejemplo si quisiéramos hacer la desestructuración de la posición 3 del arreglo y no quisiéramos poner los nombres de los elementos anteriores. Tendríamos que dejar las comas puestas en la desestructuración para indicar que no estaremos tomando esos elementos. Ejemplo:

const personajes = ['Goku', 'Vegeta', 'Trunks'];

const [ , , soloTrunks] = personajes;

console.log( soloTrunks );

Existe también la posibilidad de que necesitemos hacer la desestructuración de un elemento que aún no existe en el arreglo, pero que lo estemos esperando. La ventaja de la desestructuración de los arreglos es que también podemos poner un valor por defecto. Ejemplo:

const personajes = ['Goku', 'Vegeta', 'Trunks'];

const [ , , , Krilin = "No tiene valor" ] = personajes;

console.log( Krilin ); // No tiene valor

Pero si tuviéramos los valores en el arreglo, solamente se omitirían. Ejemplo

const personajes = ['Goku', 'Vegeta', 'Trunks', 'Krilin'];

const [ , , , Krilin = "No tiene valor" ] = personajes;

console.log( Krilin ); // Krilin

Lo mismo podemos implementar cuando estemos trabajando con los arreglos en la parte de los argumentos o retornos de una función. Ejemplo:

const informacionPersona = () =>{

    return ["Benjamín", "Ayala"]

}

const [nombre, apellido] = informacionPersona();

console.log(nombre, apellido) // Benjamín Ayala

**Desestructuración del arreglo desde los argumentos de la función.**

const informacionPersona = ([nombre, apellido]) => [nombre, apellido]

const [nombre, apellido] = informacionPersona(["Benjamín", "Ayala"]);

console.log(nombre, apellido) // Benjamín Ayala

const usarEstado = ( valor ) => {

    return [ valor , ()=>{ console.log("Hola mundo") } ];

}

const [nombre, setNombre] = usarEstado( "Gohan" )

console.log( nombre );

setNombre();