

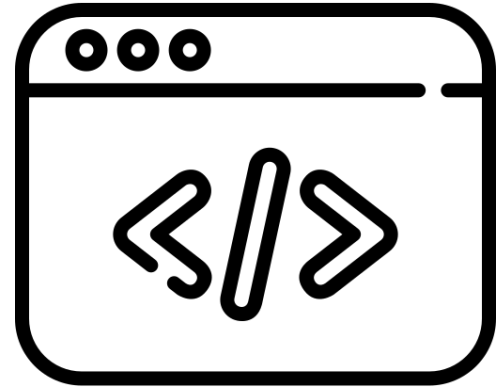
# Map y keys



# Map

La función `map` de `Array` (`Array.prototype.map`) se usa para crear un nuevo array con los resultados de llamar a una función aplicada sobre cada elemento de un array. Dicho de otra forma, `map` aplica la función que se le indique en cada uno de los elementos del array y devuelve un nuevo array de resultados.

La función `map` nos permite mapear, o transformar, los elementos de un array en otros elementos para un nuevo array.



# Map

Veamos su sintaxis en forma de función arrow:

```
map((element) => { ... } )  
map((element, index) => { ... } )
```

Y en forma de callback inline:

```
map(function callbackFn(element) { ... })  
map(function callbackFn(element, index) { ... })
```

La función map acepta un array como tercer parámetro, pero la forma en que más se utiliza solo necesita los dos primeros parámetros.

# Keys

En React la palabra clave `key` es un atributo especial de tipo `string` que se debe incluir al crear listas de elementos. React lo utiliza para identificar cuáles elementos han cambiado, se han agregado o se han eliminado.

{}

```
function MiLista(props) {  
  return (  
    <div>  
      <ol>  
        {props.items.map((item, i)=><li key={i+item}> {item}</li>)}  
      </ol>  
    </div>  
  );  
}
```

Las **keys** deben ser dadas a los elementos dentro del mapeo del **array** para darles una identidad única y estable.



Es muy importante asignar la propiedad `key` a los elementos dentro de una lista para que React pueda identificarlos con precisión. De no hacerlo, recibiremos una advertencia de que se debe proporcionar una clave para los elementos de la lista.

DigitalHouse>