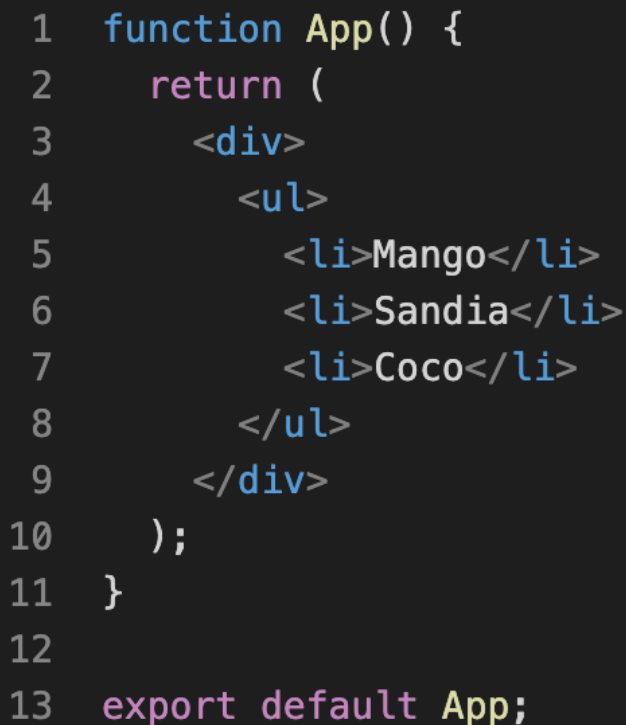


Para saber más: método map

Uno de los métodos más utilizados en React respecto a los arreglos es el método `map`. Lo interesante de este método es el hecho que nos regresa un NUEVO arreglo a partir de un arreglo base.

En el siguiente ejemplo ambos códigos generaron el mismo resultado, sin embargo, al utilizar el método `map` nos ayuda a hacer dinámico nuestro código y no repetir secciones de código innecesariamente.

En este primer ejemplo tenemos una lista desordenada con 3 ``, cada uno representando una fruta, este es un ejemplo sencillo pero imaginemos que queremos cambiar el html y en lugar de ser `` sean etiquetas `span`, tendríamos que hacerlo por cada elemento.



```
1  function App() {
2    return (
3      <div>
4        <ul>
5          <li>Mango</li>
6          <li>Sandia</li>
7          <li>Coco</li>
8        </ul>
9      </div>
10   );
11 }
12
13 export default App;
```

Haciendo uso del método `map` lo que tenemos es un código tal vez al principio más confuso pero una vez que lo entiendas verás qué útil es. Este método SOLO está disponible para arreglos, en este caso el arreglo de frutas, por cada elemento

("fruta") que se encuentra en el arreglo se ejecutará el arrow function que se definió y por cada elemento se va a crear un nuevo.

```
1  const frutas = ["Mango", "Sandia", "Coco"];
2
3  function App() {
4    return (
5      <div>
6        <ul>
7          {frutas.map((fruta) => {
8            return <li>{fruta}</li>;
9          })}
10       </ul>
11     </div>
12   );
13 }
14
15 export default App;
16
```

Te dejo algunos enlaces donde podrás encontrar más información.

- [Array.prototype.map\(\)](#)
- [JavaScript Array map\(\)](#)
- [React ES6 Array Methods](#)