

PRÁCTICA 5.3

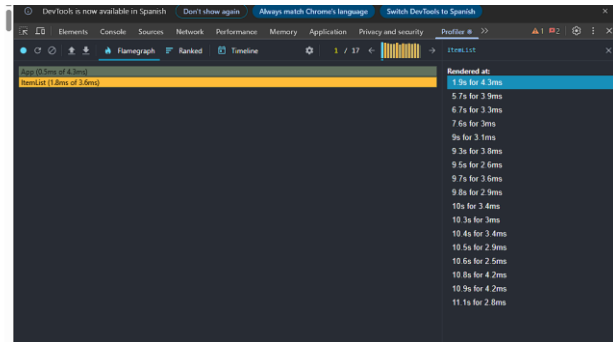
Pedro Vázquez Tévar

En la parte derecha podemos ver los tiempos de cada render, es decir, cada vez que pulsamos el boton ‘Incrementar’.

Profiling en React

Incrementar (17)

mm 0
mm 1
mm 2
mm 3
mm 4
mm 5
mm 6
mm 7
mm 8
mm 9
mm 10
mm 11
mm 12
mm 13
mm 14
mm 15
mm 16



Si usamos Memoización:

```

1 | import { FC, memo, useState, useMemo } from "react";
2 |
3 | // Props para ItemList
4 | type ItemListProps = {
5 |   items: string[];
6 | };
7 |
8 | // ItemList memoizado
9 | Complexity is 5 Everything is cool!
10 | export const ItemList: FC<ItemListProps> = memo(({ items }) => {
11 |   console.log("ItemList render"); // Para comprobar si se re-renderiza
12 |   return (
13 |     <div>
14 |       {items.map((item, index) => (
15 |         <div key={index}>{item}</div>
16 |       ))}
17 |     </div>
18 |   );
19 | });
20 | Complexity is 10 It's time to do something...
21 | export const App: FC = () => {
22 |   const [count, setCount] = useState<number>(0);
23 |   // Memoizamos la lista de items para que no se regenere en cada render
24 |   const items = useMemo(
25 |     () => Array.from({ length: 1000 }, (_, i) => `Item ${i}`),
26 |     []
27 |   );
28 |
29 |   return (
30 |     <div>
31 |       <h1>Profiling en React</h1>
32 |       <button onClick={() => setCount((prev) => prev + 1)}>
33 |         Incrementar ({count})
34 |       </button>
35 |       <ItemList items={items} />
36 |     </div>
37 |   );
38 | };
39 |

```

Profiling en React

Incrementar (4)

Item 0
Item 1
Item 2
Item 3
Item 4
Item 5
Item 6
Item 7
Item 8
Item 9
Item 10
Item 11
Item 12
Item 13
Item 14
Item 15
Item 16
Item 17
... ..

