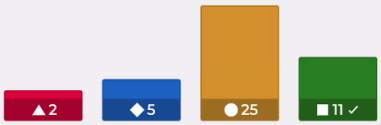
React Hooks es...

Next

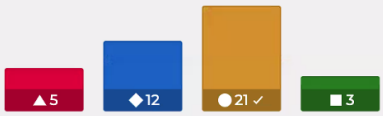


Show media

- ▲ Ninguna de las anteriores. ✖
- ◆ La nueva versión de React. ✖
- ● Una forma alternativa que propone React al manejo de estado(s). ✖
- ■ El paradigma de programación funcional que propone React. ✔

¿ Qué tipo de dato es el STATE ?

Next



Show media

- ▲ Array ✖
- ◆ Todas las anteriores. ✖
- ● Object ✔
- ■ Primitive ✖

¿ Para qué necesitamos desestructurar useState ?

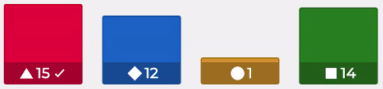
Next

Show media

- ▲ Ninguna de las anteriores. ✖
- ◆ Para evitar declarar dos VAR con valores que provienen del mismo lugar. ✔
- ● Si se declara de otra manera, no funciona. ✖
- ■ Por convención de sintaxis. ✖

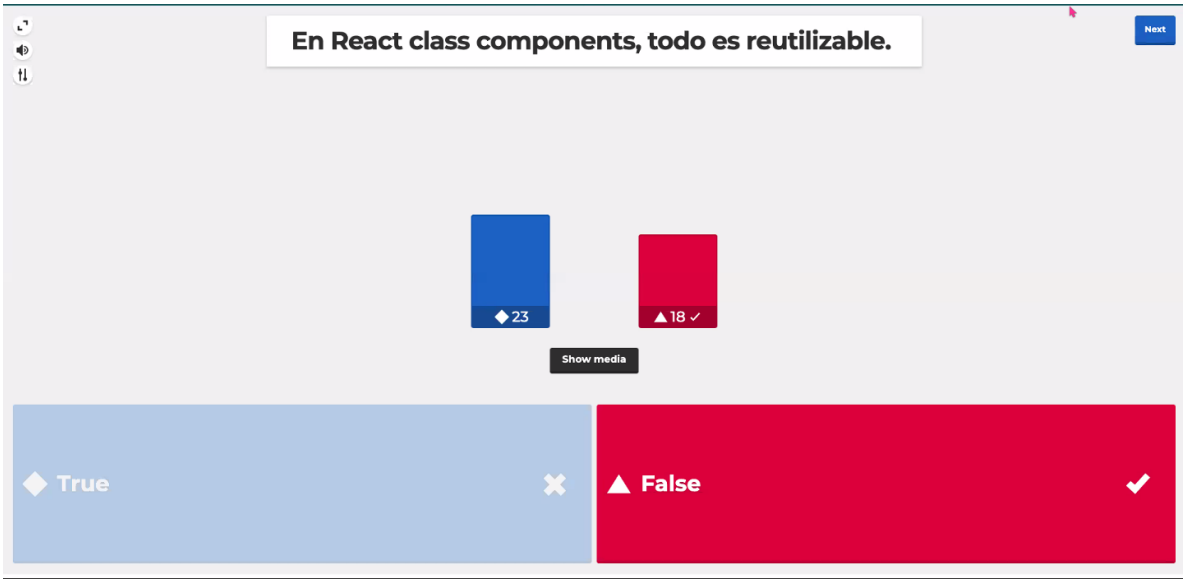
¿ Qué soluciones viene a proponer React Hooks ?

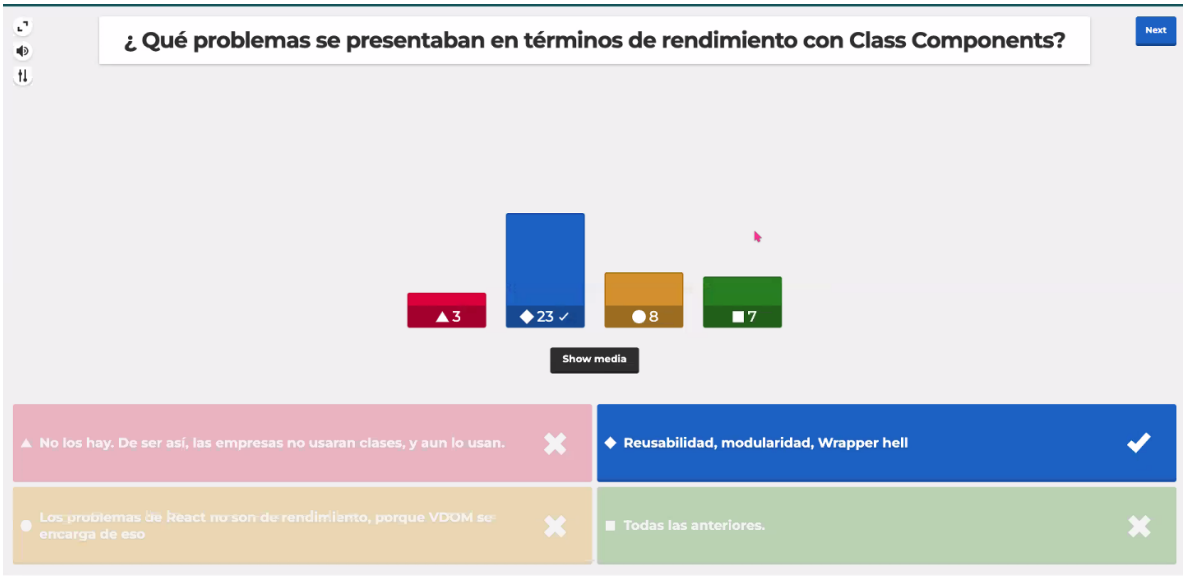
Next



Show media

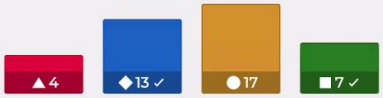
- ▲ Permite la modularidad de códigos, especialmente la relaciona a funciones. ✔
- ◆ Mejora el rendimiento al evitarnos renders innecesarios. ✖
- ● No soluciona nada. Solo es una versión nueva que reduce lineas de código. ✖
- ■ Todas las anteriores. ✖


En React class components, todo es reutilizable.
Next
◆ 23
▲ 18 ✓
Show media
◆ True
▲ False


¿ Qué problemas se presentaban en términos de rendimiento con Class Components?
Next
▲ 3
◆ 23 ✓
● 8
■ 7
Show media
▲ No los hay. De ser así, las empresas no usaran clases, y aun lo usan.
◆ Reusabilidad, modularidad, Wrapper hell
● Los problemas de React no son de rendimiento, porque VDOM se encarga de eso
■ Todas las anteriores.

Que la estructura de React se jerárquico, y no hereditario...

Next



Show media

▲ Ninguna de las anteriores. ✖

◆ Significa que no hereda los métodos. Por ende la lógica no es reutilizable. ✔

● Significa que un componente es padre de otro componente. ✖

■ Significa que los métodos de la clase padre no pueden ser heredados. ✔

¿Qué diferencia hay entre un método de ciclo de vida, y un método?

Next

Show media

▲ Todas las anteriores. ✔

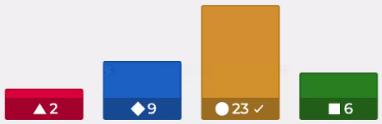
◆ Los ciclos de vida manejan efectos secundarios. ✖

● Los métodos son declarados por el programador. ✖

■ Ciclos de vida no están presentes en Functional componentes. ✖

¿ Qué se entiene como efecto secundario ?

Next



Show media

- ▲ Ninguna de las anteriores. ✖
- ◆ A los triggers de los eventos. ✖
- ● Cualquier comportamiento que se de a partir del scope de una función. ✔
- ■ A la lógica que existe dentro de un método. ✖

¿ Qué hook se encarga del ciclo de vida de un componente ?

Next

Show media

- ▲ useState junto con useEffect ✖
- ◆ useState ✖
- ● useEffect y useLayoutEffect ✔
- ■ ComponentDidUpdate() ✖


¿ Cuántos argumentos toma useEffect ?
Next
▲2 ✓
◆20 ✓
●9
■4
Show media
▲ useEffect
◆ Dos: una función flecha, y un arreglo vacío llamado arreglo de dependencia.
● Tres: Una función flecha, el arreglo de dependencias, y un return vacío.
■ El hook que tiene argumentos es el useState, que se desestructura.