



Curso Práctico de React.js



Aprobaste el curso

Felicitaciones, ya puedes acceder a tu [diploma digital](#)

10

Calificación

15 / 15

Aciertos

1. Si nos basamos en el patrón MVC (modelo-vista-controlador), ¿de cuál se encarga React.js?

Vista



2. React.js nos permite construir interfaces con base en:

Componentes



3. ¿Cuál es la súper empresa encargada de construir React.js?

Facebook



4. ¿Qué es JSX?

El sistema de plantillas de React que combina la sintaxis de JavaScript con XML.



5. ¿Qué es el Virtual DOM?

Una copia o representación del DOM para evitar cambios innecesarios en el DOM real.



6. ¿Cómo creamos un elemento de React que se transforme en un DIV con la clase "container" usando JSX?

```
<div className="container"></div>
```



7. ¿Cuáles son las herramientas INDISPENSABLES que debemos instalar para usar React en una aplicación web?

react y react-dom



8. ¿Cómo renderizamos el componente principal de nuestra aplicación con React 17?

```
ReactDOM.render()
```



9. Escalar nuestro proyecto será mucho más fácil si usamos Create React App en vez de configurar manualmente Webpack y Babel. Esta afirmación es:

Falsa



10. Teniendo lo siguiente:

```
const App = () => {  
  return (  
    <Layout>  
      <Login />  
    </Layout>  
  );  
}
```

¿Cuál es el código correcto del componente Layout?



```
const Layout = ({ children }) => { return (  
  
  <div classname="Layout">{children}</div>  
  
  ); }
```

11. ¿Qué es React Router DOM?

La herramienta más popular de React para separar nuestra aplicación entre páginas / rutas.



12. ¿Cuál es la forma correcta crear un 404 con React Router DOM 5?

Dentro de los componentes BrowserRouter y Switch. De último lugar con las demás rutas de la aplicación. Y con <Route path="/" component={NotFound} />.



13. ¿Qué herramienta de React nos permite trabajar con el estado en nuestros componentes?

React.useState



14. ¿Qué herramienta de React nos permite compartir un estado con todos los componentes de la aplicación?

React.Context



15. ¿Cuál es la forma correcta de actualizar un estado cuando los usuarios den click en un botón?



```
const Button = () => {  
  const [count, setCount] = React.useState(1);  
  return (  
    <button onClick={() => setCount(count + 1)}>  
      ¡Click!  
    </button>  
  );  
}
```

REGRESAR