

# Intro a React:

## Descubre su potencial



Javier Diaz

[twitch.tv/coderdiaz](https://twitch.tv/coderdiaz)

[@coderdiaz](https://twitter.com/coderdiaz)



## Agenda - ¿Qué aprenderemos?

- ✓ ¿Qué es React?
- ✓ Su importancia en el mundo del frontend
- ✓ Cómo funciona y sus características
- ✓ createElement vs JSX
- ✓ Prepara tu ambiente de trabajo
- ✓ Arquitectura de componentes
- ✓ Props y manejo de eventos.



# ¿Qué es React?

Es una biblioteca open-source para crear interfaces de usuario creado por Jordan Walke en 2011 durante su estancia en Facebook



# ¿Por qué debería aprenderlo?

- ✓ Curva de aprendizaje
- ✓ Te ayuda a cambiar tu modelo mental
- ✓ Compatibilidad entre versiones
- ✓ Alta demanda en el mercado
- ✓ Fácil adopción y una muy buena experiencia para el dev

Recomendaciones:

<https://youtu.be/8pDqJVdNa44>

@coderdiaz



# Cosas con las que deberías tener cuidado:

- ⚠ No es opinado
- ⚠ Es fácil caer en océano de problemas
- ⚠ Exige un conocimiento solido de JavaScript
- ⚠ JSX a veces es un dolor de @%\$!\*%





# Como funciona



# Entendiendo la web

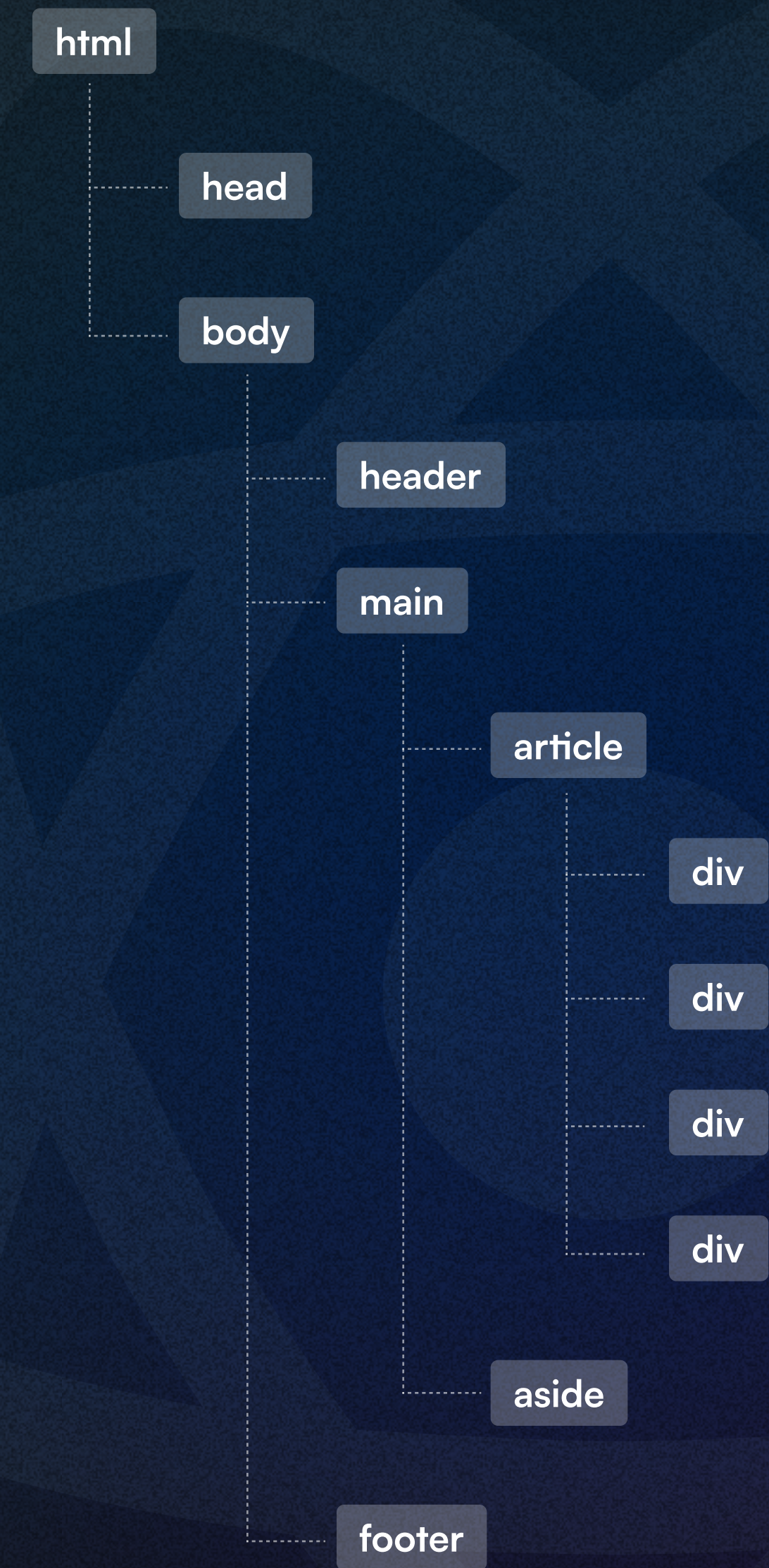


```
1  <html>
2    <head></head>
3    <body>
4      <header></header>
5      <main>
6        <article>
7          <div></div>
8          <div></div>
9          <div></div>
10         <div></div>
11        </article>
12        <aside></aside>
13      </main>
14      <footer></footer>
15    </body>
16  </html>
```



# Interpretando nuestro HTML (DOM)

```
1  <html>
2    <head></head>
3    <body>
4      <header></header>
5      <main>
6        <article>
7          <div></div>
8          <div></div>
9          <div></div>
10         <div></div>
11        </article>
12      <aside></aside>
13    </main>
14    <footer></footer>
15  </body>
16 </html>
```





# Entendiendo a React



```
1 <html>
2   <head></head>
3   <body>
4     <div id="app"></div>
5   </body>
6 </html>
```



# Interpretando nuestro HTML

(DOM)

```
1 <html>
2   <head></head>
3   <body>
4     <div id="app"></div>
5   </body>
6 </html>
```





# Montando la aplicación con React

(Virtual DOM)

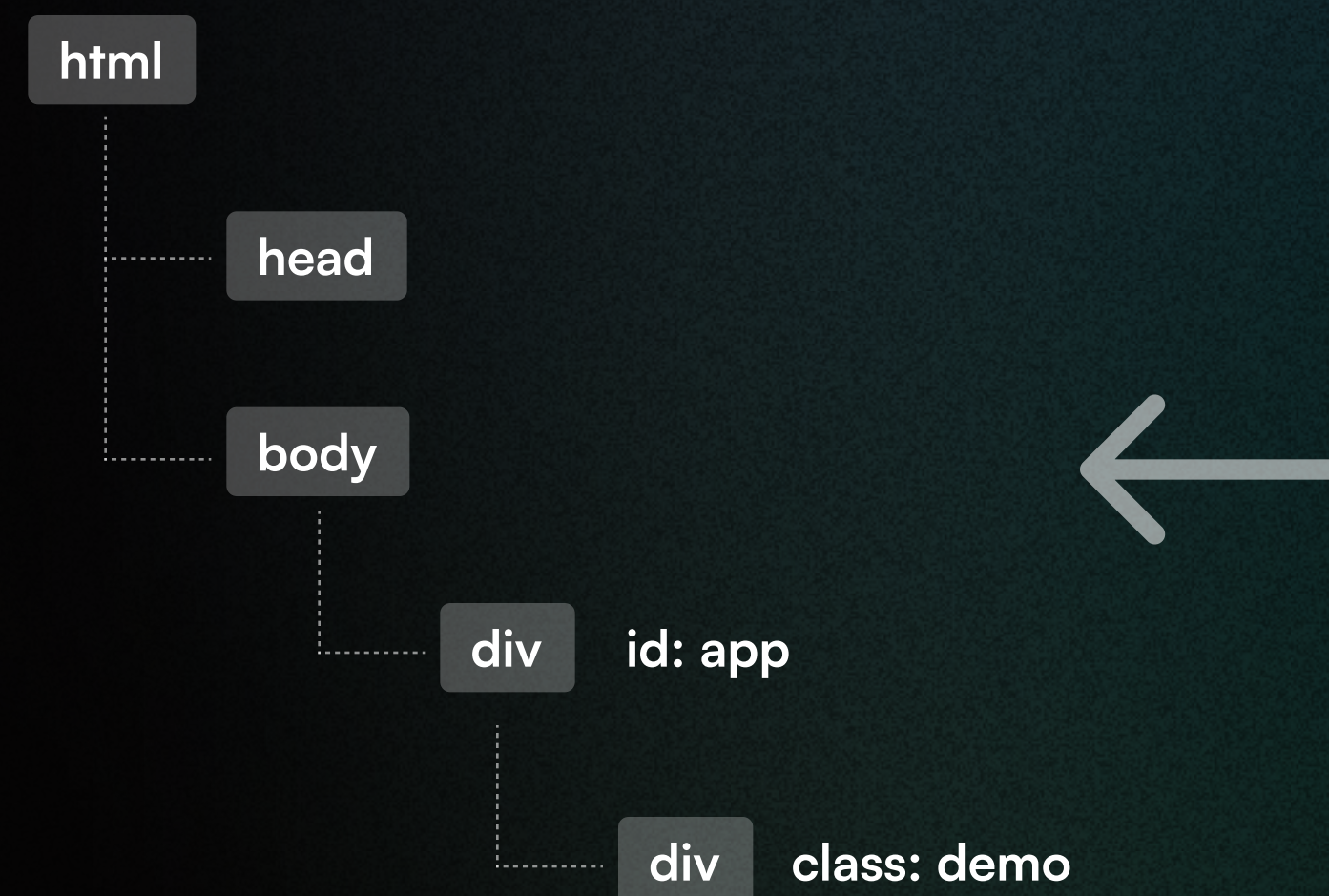
```
1 <html>
2   <head></head>
3   <body>
4     <div id="app"></div>
5   </body>
6 </html>
```



```
1 // index.js
2 import { render } from 'react-dom';
3
4 const root = document.getElementById('app');
5 render(<div className="demo" />, root);
```

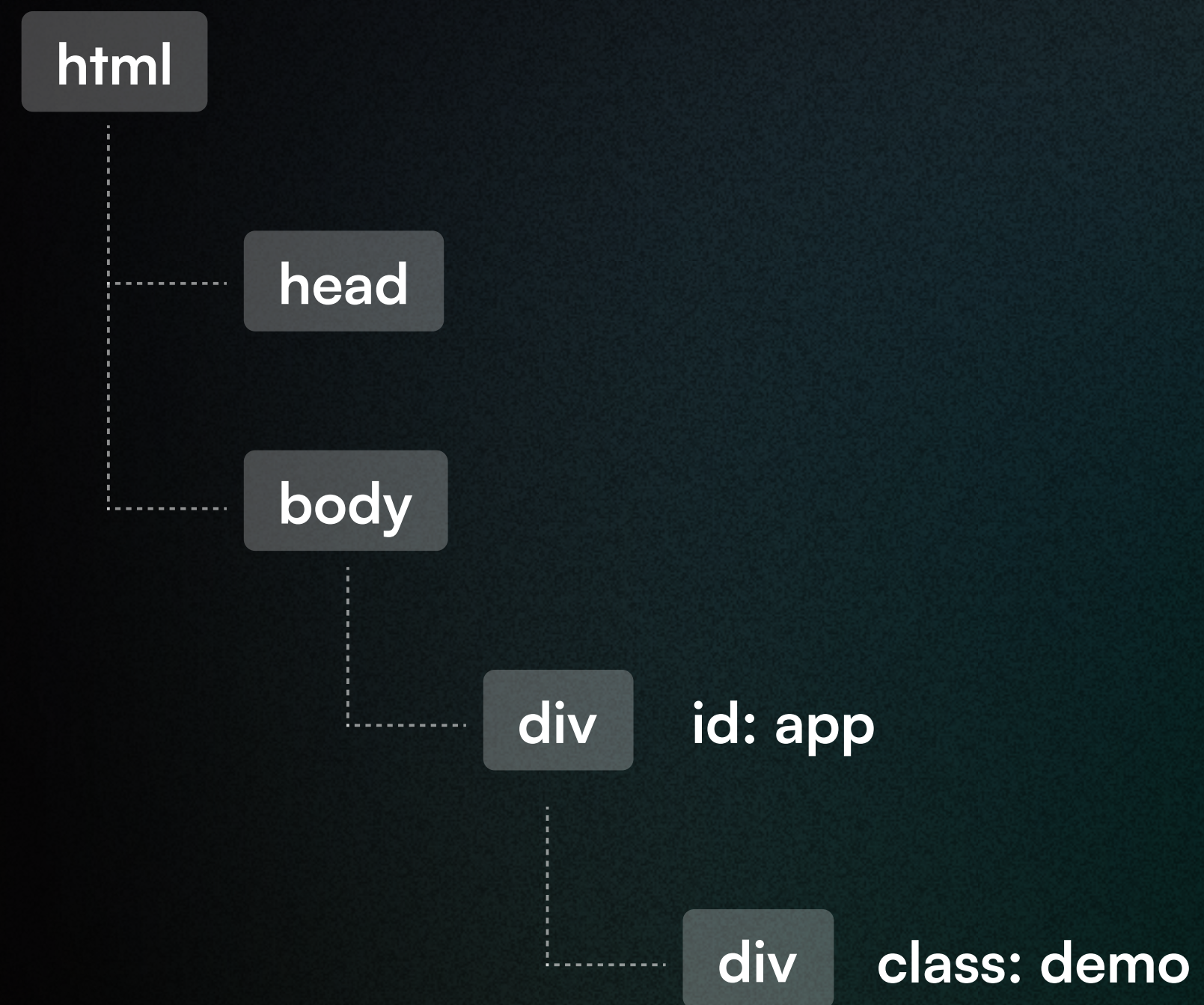
Ejecutamos el rendering de React

Virtual DOM





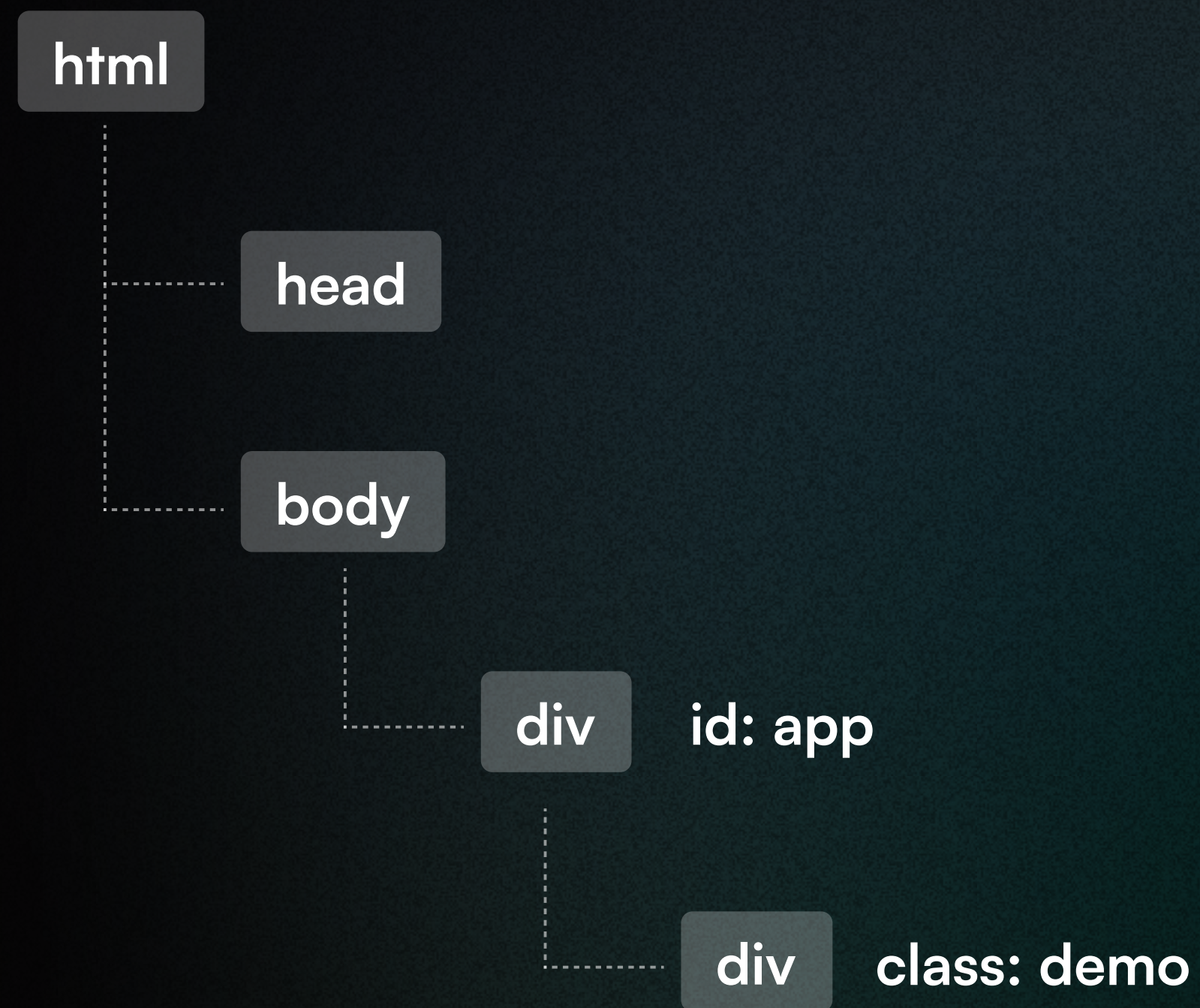
# VDOM



# Real DOM



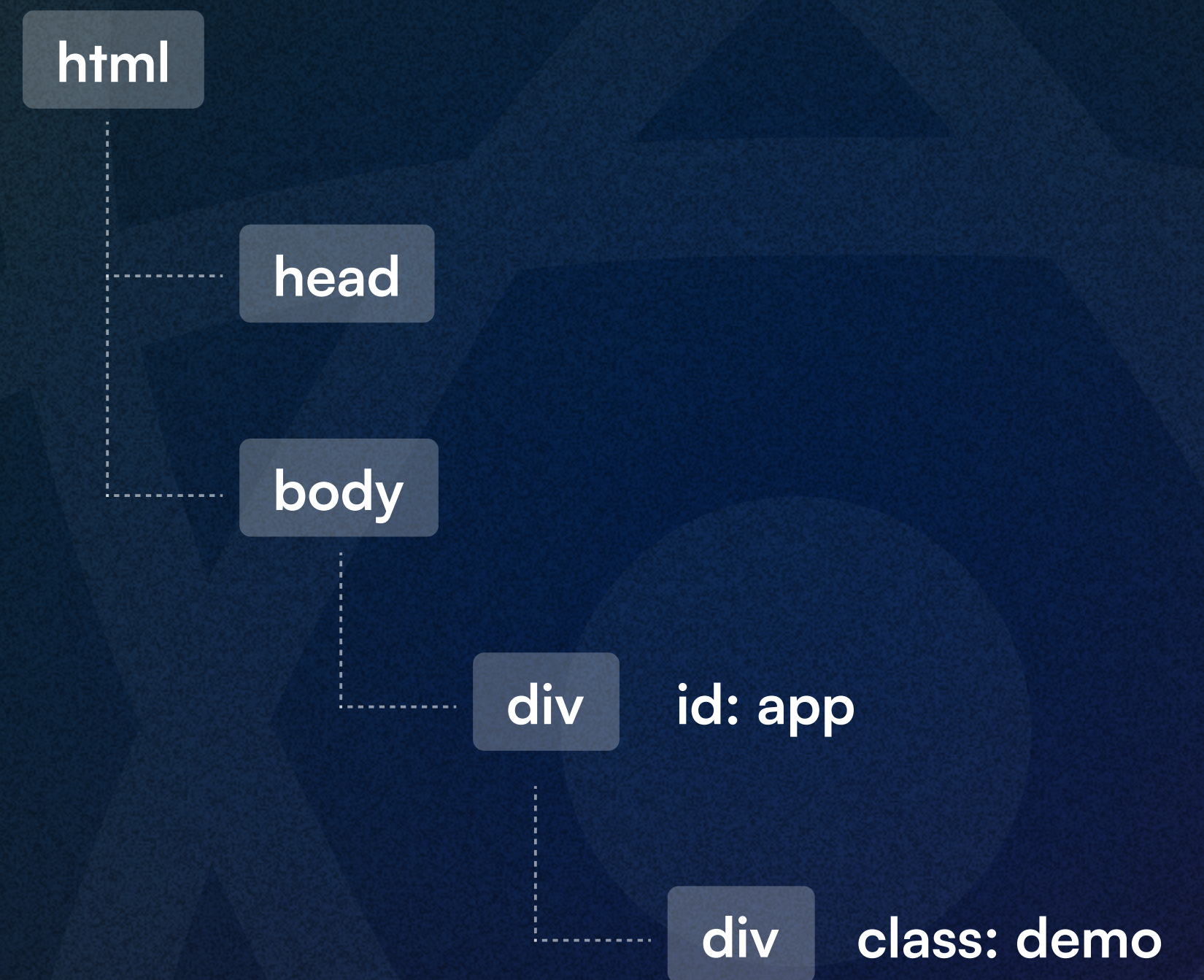




react-dom

→

# Reconciliación



Lectura:

<https://legacy.reactjs.org/docs/reconciliation.html>

@coderdiaz



# Características

🎲 Declarativo

🎲 Unidireccional

🎲 Basado en componentes

🎲 Virtual DOM

🎲 Versátil

```
// Declarativo  
const element = <h1>Hello, world</h1>
```

```
// Imperativo  
const element = document.createElement('h1')  
element.innerHTML = 'Hello, world'
```



# createElement vs JSX



# createElement

Es la forma programática con la que puedes crear elementos de React y en lo cuál nuestros componentes escritos en JSX son traducidos.

```
import React from 'react'
import ReactDOM from 'react-dom'

// Create a new component
const element = React.createElement(
  'h1',
  { className: 'my-class' },
  'Hola, mundo!'
)

const root =
  document.getElementById('root')

ReactDOM.render(element, root)
```

Lectura recomendada:

<https://react.dev/reference/react/createElement>

@coderdiaz



# JSX

Es una sintaxis más intuitiva y legible recomendada para escribir componentes en React.

Combina HTML y JavaScript en el mismo archivo.

```
import React from 'react'
import ReactDOM from 'react-dom'

// Create a new component
const element = () => {
  return (
    <h1 className="my-class">
      Hola, mundo!
    </h1>
  )
}

const root =
  document.getElementById('root')

ReactDOM.render(element, root)
```

Lectura recomendada:

<https://react.dev/learn/writing-markup-with-jsx>

@coderdiaz



# Pensando en componentes



# Observa la siguiente estructura

🤔 ¿Cuántos componentes ves?

Search

Your products

## Fruits

Apple	\$1
Dragonfruit	\$1
Watermelon	\$3

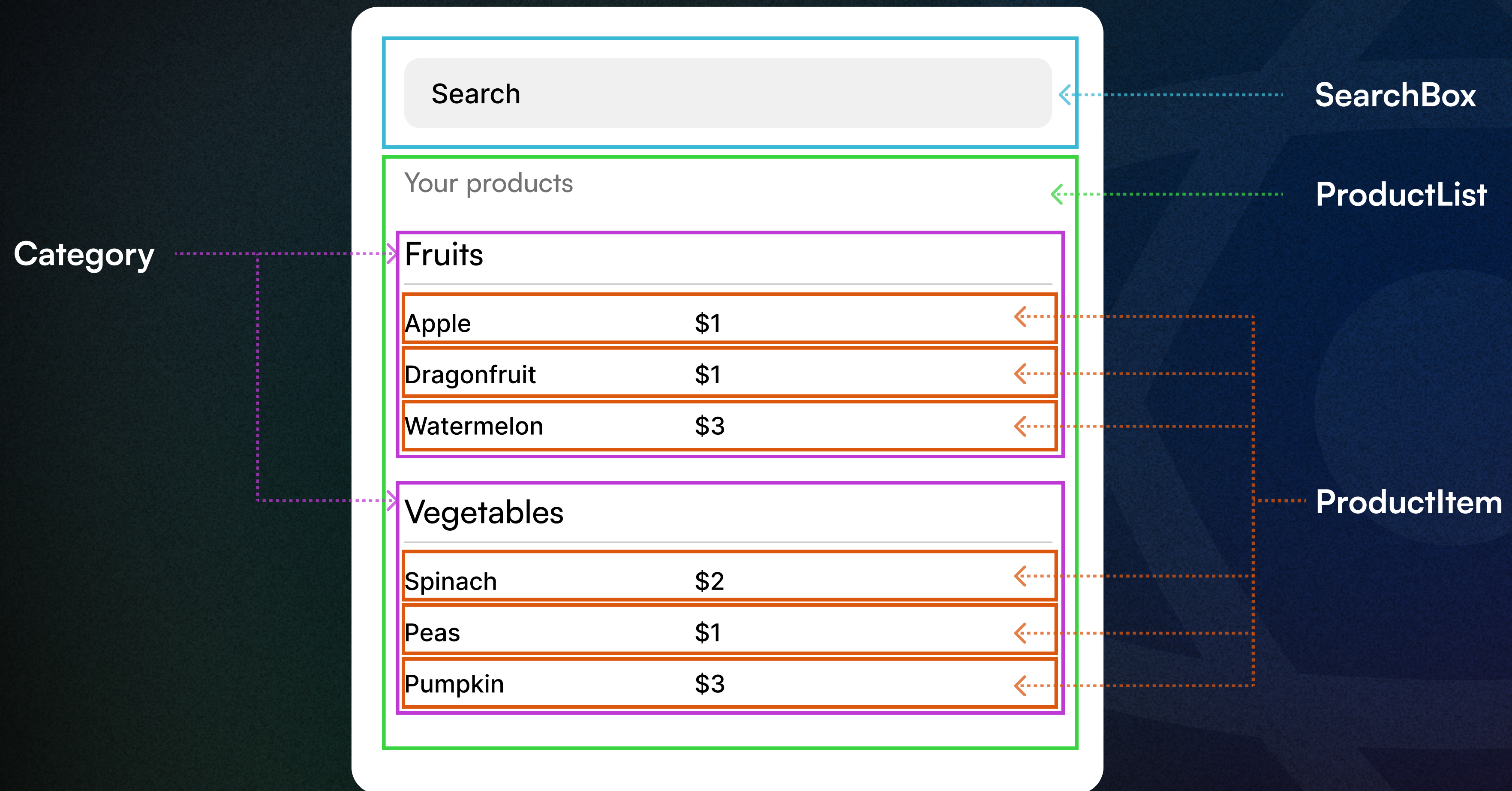
## Vegetables

Spinach	\$2
Peas	\$1
Pumpkin	\$3



# Observa la siguiente estructura


🤔 ¿Cuántos componentes ves?






Observa la siguiente estructura


🤔 ¿Cuántos componentes ves?

 **Route (7)** Show rote

Cargo loading (2)




**6375 Bren Mar Dr, Alexandria, VA 22312**  
🕒 7 am, Mon 09/05/2022 1st loading 1




**6025 Farrington Ave Unit B, Alexandria, VA 22304**  
🕒 8 am, Mon 09/05/2022 2nd loading


Unloading cargo (5)




**5824 Rising Sun Ave, Philadelphia, PA 19120**  
🕒 2:00 pm Mon 09/05/2022 1st unloading




**2417 Welsh Rd, Philadelphia, PA 19114**  
🕒 3:00 pm Mon 09/05/2022 2nd unloading



**630 Main St, Hackensack, NJ 07601**  
🕒 3:30, Mon 09/05/2022 3rd unloading



**400 Huyler St, South Hackensack, NJ 07606**  
**Goodwill NYNJ Outlet Store & Donation Center**  
🕒 16:00 Mon 09/05/2022 4th unloading



**665 American Legion Dr, Teaneck, NJ 07666**  
🕒 17:00 Mon 09/05/2022 5th unloading

Change addresses



# Observa la siguiente estructura

🤔 ¿Cuántos componentes ves?

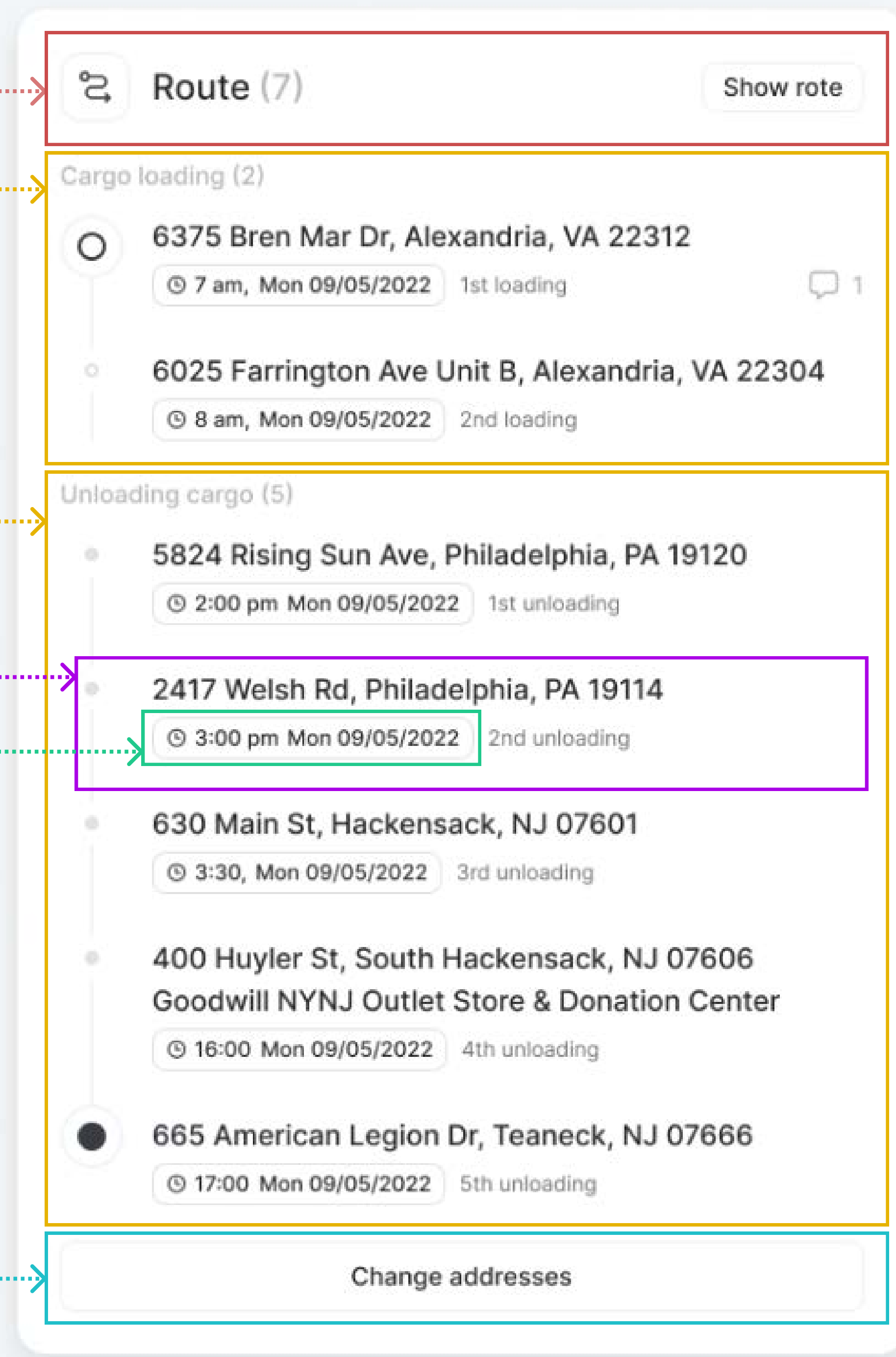
Heading

BlockSection

CargoItem

DateFormat

Button





# Props

Son propiedades con las cuales podemos pasar datos de una componente a otro.

Estos son inmutables, lo que significa que no se pueden modificar una vez dentro.

```
const ProductItem = ({ name, price }) => {  
  return (  
    <div className="product__item">  
      <span className="product__name">  
        {name}  
      </span>  
      <span className="product__price">  
        {price}  
      </span>  
    </div>  
  )  
}
```

Lectura recomendada:

<https://react.dev/learn/passing-props-to-a-component>

@coderdiaz



# Manejo de eventos

Al igual que JavaScript con HTML, puedes agregar interacción a tus componentes.

## ¿Qué es un evento?

Son interacciones que suceden sobre elementos dentro de nuestra página. Cuando usamos JavaScript en nuestra web, JavaScript puede “reaccionar” ante las interacciones que el usuario tiene con ella.

Lectura recomendada:

<https://react.dev/learn/responding-to-events>

```
const Button = () => {  
  const handleClick = () => {  
    console.log('clicked!')  
  }  
  
  return (  
    <button onClick={handleClick}>  
      Click me  
    </button>  
  )  
}
```



# Sígueme en X como



@coderdiaz

# Muchas gracias!

[twitch.tv/coderdiaz](https://twitch.tv/coderdiaz)

