

## ÍNDICE

[Parte A: Estilo en línea \(inline styles\):](#)

[Parte B: Archivos CSS \(global o por componente\):](#)

[Parte C: CSS Modules \(alcance local por defecto\):](#)

[Parte D: Tailwind CSS \(utilidades\):](#)

[Miniretillo:](#)

[Preguntas:](#)

## Parte A: Estilo en línea (inline styles):

The screenshot shows a browser window displaying the text "Estilos en línea en React" and three paragraphs with different styles. To the right is a code editor showing the corresponding `App.jsx` file:

```
src > App.jsx > App > highlightedStyle > paragraphStyle
1 function App() {
2   const highlightedStyle = (
3     ...paragraphStyle,
4     color: 'red'
5   )
6
7   return (
8     <>
9       <h1 style={{ color: 'blue', textAlign: 'center' }}>
10      Estilos en línea en React
11    </h1>
12
13     <p style={paragraphStyle}>Primer párrafo</p>
14     <p style={paragraphStyle}>Segundo párrafo</p>
15
16     <p style={highlightedStyle}>Párrafo con estilo extendido</p>
17   </>
18 }
19
20
21
22
23
24
25
26
27 export default App
```

(Captura con código propio, elaboración propia)

## Parte B: Archivos CSS (global o por componente):

The screenshot shows a browser window displaying the text "CSS externo en React" and two paragraphs with different classes. To the right is a code editor showing the corresponding `App.jsx` file:

```
src > App.jsx > # style.css U > App > style.css
1 import './style.css';
2
3 function App() {
4   return [
5     <>
6       <h1>CSS externo en React</h1>
7
8       <p className="paragraph-text">
9         Párrafo con CSS externo
10      </p>
11
12       <p className="paragraph-text">
13         Otro párrafo reutilizando la clase
14      </p>
15   </>
16 }
17
18
19 export default App
```

(Captura con código propio, elaboración propia)

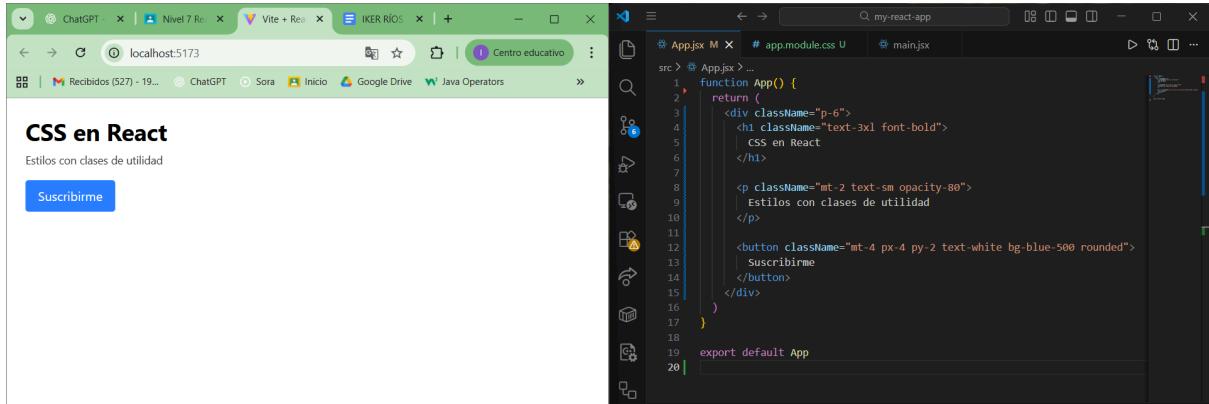
## Parte C: CSS Modules (alcance local por defecto):

The screenshot shows a browser window displaying the text "CSS Modules en React" and two paragraphs with different styles. To the right is a code editor showing the corresponding `App.jsx` file:

```
src > App.jsx > # app.module.css U > App > app.module.css
1 import styles from './App.module.css';
2
3 function App() {
4   return [
5     <>
6       <h1>CSS Modules en React</h1>
7
8       <p className={`${styles.baseText} ${styles.primary}`}>
9         Texto azul con tamaño base
10      </p>
11
12       <p className={`${styles.baseText} ${styles.secondary}`}>
13         Texto verde con tamaño base
14      </p>
15   </>
16 }
17
18
19 export default App
```

(Captura con código propio, elaboración propia)

## Parte D: Tailwind CSS (utilidades):

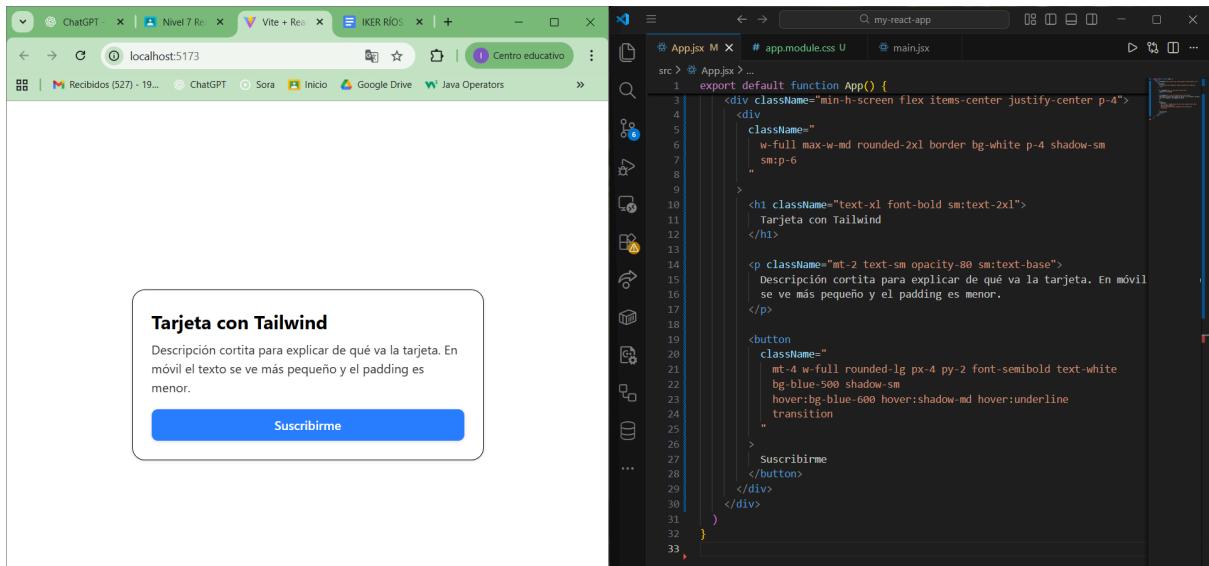


The screenshot shows a browser window with the URL `localhost:5173`. The page title is "Centro educativo". The content displays a heading "CSS en React" and a button labeled "Suscríbeme". The code editor on the right shows the corresponding `App.jsx` file:

```
src > App.jsx < # app.module.css U main.jsx
1 function App() {
2   return (
3     <div className="p-6">
4       <h1 className="text-3xl font-bold">
5         | CSS en React
6       </h1>
7
8       <p className="mt-2 text-sm opacity-80">
9         | Estilos con clases de utilidad
10      </p>
11
12      <button className="mt-4 px-4 py-2 text-white bg-blue-500 rounded">
13        | Suscríbeme
14      </button>
15    </div>
16  )
17
18  export default App
19
20
```

(Captura con código propio, elaboración propia)

## Miniretillo:



The screenshot shows a browser window with the URL `localhost:5173`. The page title is "Centro educativo". The content displays a card titled "Tarjeta con Tailwind" with a description and a "Suscríbeme" button. The code editor on the right shows the corresponding `App.jsx` file:

```
src > App.jsx < # app.module.css U main.jsx
1 export default function App() {
2   return (
3     <div className="min-h-screen flex items-center justify-center p-4">
4       <div
5         className="w-full max-w-md rounded-2xl border bg-white p-4 shadow-sm sm:p-6"
6       >
7
8         <h1 className="text-xl font-bold sm:text-2xl">
9           | Tarjeta con Tailwind
10          </h1>
11
12         <p className="mt-2 text-sm opacity-80 sm:text-base">
13           Descripción cortita para explicar de qué va la tarjeta. En móvil se ve más pequeño y el padding es menor.
14         </p>
15
16         <button
17           className="mt-4 w-full rounded-lg px-4 py-2 font-semibold text-white bg-blue-500 shadow-sm hover:bg-blue-600 hover:shadow-md hover:underline transition"
18           >
19             | Suscríbeme
20           </button>
21         </div>
22       </div>
23     </div>
24   )
25
26
27
28
29
30
31
32
33
```

(Captura con código propio, elaboración propia)

**Preguntas:**

**¿Qué diferencia hay entre onClick y onSubmit?**

onClick se ejecuta al hacer clic en un elemento; onSubmit se usa para enviar formularios y captura también la tecla Enter.

**¿Por qué usamos e.preventDefault() en un formulario?**

Se usa para evitar el comportamiento por defecto del formulario (recarga de página) y gestionar el envío con React.

**¿Qué es una "entrada controlada" y por qué usamos value + onChange?**

Es un input cuyo valor depende del estado de React; usamos value y onChange para tener control total del dato.

**En tu mini-reto, que estado(s) manejas y que evento(s) los actualizan?**

No se usan estados ni eventos, ya que solo hay estilos y comportamiento visual (hover).