

React

Introducción a componentes

“

Los **componentes** permiten separar la interfaz de usuario en **piezas independientes reutilizables** pensando cada pieza de forma aislada.

Técnicamente son **funciones** nativas de **JavaScript**.



REACT

Componentes stateless

Se llaman componentes **stateless** (componentes funcionales o componentes sin estado) porque internamente implementan un **return con lógica sencilla** que entrega una estructura **html**.

Dentro de la lógica sencilla eventualmente podemos encontrar estructuras de control (“if”) y seguramente varios métodos map para iterar arrays.

```
function Navbar() {  
  return (  
    <nav>  
      <a href="/home">Home</a>  
      <a href="/productos">Productos</a>  
    </nav>  
  );  
}
```

Los componentes **ÚNICAMENTE** pueden devolver 1 sólo elemento html.

Dentro del **return** no puede haber etiquetas “hermanas”.

REACT

Crear un componente statless

Crea un archivo con extensión `.js` con el nombre de la función que llevará dentro. En este caso: `/src/components/Navbar.js`

```
{ }
```

```
import React from "react";
```

```
{ }
```

```
function Navbar() {  
  return (  
    <nav>  
      <a href="/home">Home</a>  
      <a href="/productos">Productos</a>  
    </nav>  
  );  
}
```

```
{ }
```

```
Export default Navbar;
```

REACT

Implementar un componente statless

En el archivo **App.js** importamos el componente **Navbar**.

Luego lo implementamos como si fuese una **etiqueta de HTML** **dentro** de la estructura **del return** del componente App.

No olvides poner una barra (/) al final para cerrar la etiqueta.

```
{ } import Navbar from “./components/Navbar”;
```

```
{ } function App() {  
  return (  
    <div className=“App”>  
      <Navbar />  
    </div>  
  );  
}
```

REACT

Introducción a componentes

Para saber más sobre componentes puedes acceder a la documentación oficial de React haciendo click en el siguiente [Link](#):

