wawiwa
React
2022 © Wawiwa Tech | Confidential


wawiwa
6 - Componentes y Props (propiedades)
visual design
2022 © Wawiwa Tech | Confidential

https://platform.blossom-kc.com/wawiwa_tech/Launcher?assignment=38867&ref=https%3A%2F%2Fplatform.blossom-kc.com%2Fwawiwa_tech%2F...

14/11/24, 9:20 p.m.

wawiwa

¿Qué son los componentes en React?

Un componente es uno de los elementos fundamentales de React

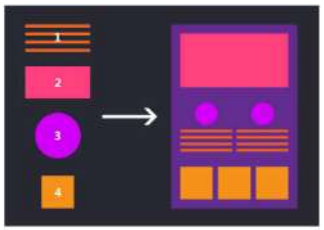
En React, desarrollas tus aplicaciones creando componentes (piezas) reusables que puedes imaginarlas como bloques de Lego.

Conceptualmente, los componentes son como funciones de JavaScript, aceptan inputs arbitrariamente (llamadas “props”) y retornar elementos de React que describen lo que debería de aparecer en la pantalla.

Puedes dividir la interfaz del usuario en componentes y trabajar con cada uno individualmente antes de combinarlos en el componente padre, que será el UI final.

Recuerda que la ley de los componentes de React - cada componente tiene **un** rol!

Y si tiene muchos, ¡entonces deben ser separados en más componentes!



PDF.js viewer

wawiwa

Tipos de componentes de React

React tiene dos tipos de componentes:

1. **Function (Función)** -

   El componente funcional es creado escribiendo una **función**.

1. **Basado en clase (Class-based)** -

   Un componente de clase se crea usando la sintaxis **ES6 class**.

https://platform.blossom-kc.com/wawiwa_tech/Launcher?assignment=38867&ref=https%3A%2F%2Fplatform.blossom-kc.com%2Fwawiwa_tech%2F...

14/11/24, 9:20 p.m.

wawiwa

Componente de Función (Function Component)

La manera más fácil de definir un componente es escribir una función Javascript:

```
function Welcome(props) {
return <h1>Hello, {props.name}</h1>; }
```

El tipo de componente **recomendado** en React es “functional component”.
Un componente funcional es básicamente una función JavaScript/ES6 que **returna un elemento React (JSX)**.

Acepta un **solo** objeto "props" (que significa properties (propiedades)) como argumento con datos.

PDF.js viewer

wawiwa

Componente de función como función de flecha

Puedes también crear un componente funcional con la definición de función de flecha:

```
const Welcome = (props) => {
  return <h1>Hello, {props.name}</h1>;
}
```

https://platform.blossom-kc.com/wawiwa_tech/Launcher?assignment=38867&ref=https%3A%2F%2Fplatform.blossom-kc.com%2Fwawiwa_tech%2F...

wawiwa

Crea un componente de función

Ubicación del componente:

Podemos crear un componente donde sea:

1. Dentro del archivo actual
2. Como otro archivo

PDF.js viewer

wawiwa

Crea un componente de función

1. Dentro del archivo actual:

```
function App() {
const Welcome = (props) => {
  return <h1>Hello, {props.name}</h1>;
}
  return (
    <h3>
      <Welcome name={"Joe"} />
    </h3>
  )
}

export default App;
```

PDF.js viewer

https://platform.blossom-kc.com/wawiwa_tech/Launcher?assignment=38867&ref=https%3A%2F%2Fplatform.blossom-kc.com%2Fwawiwa_tech%2F...

wawiwa

Ejercicio 2

Crea un componente funcional que sea responsable de un ticket de un pasajero.

Los componentes reciben estos parámetros:

- Name
- Destination
- Gender (Mr/Mrs)
- Seat

El componente retornará un div con estos detalles.

Renderiza este componente dos veces, cada vez con información distinta.

For example:

Ticket Details

Name: John Doe

Destination: New York

Gender: Mr

Seat: 14A

Ticket Details

Name: Alex Doe

Destination: Tel Aviv

Gender: Mr

Seat: 17A

PDF.js viewer

wawiwa

Crea un componente de función

2. Como otro archivo

Welcome.js U

src > Welcome.js > default

```
1  function Welcome(props) {
2      return <h1>Hello, {props.name}</h1>;
3  }
4
5  export default Welcome;
```

App.js M

src > App.js > default

```
1  import Welcome from "./Welcome";
2
3  function App() {
4    return (
5      <h3>
6        <Welcome name={"Joe"} />
7      </h3>
8    )
9  }
10
11 export default App;
```

Para poder usar un componente más tarde, lo necesitas exportar y después importarlo.

Llama al componente

PDF.js viewer

https://platform.blossom-kc.com/wawiwa_tech/Launcher?assignment=38867&ref=https%3A%2F%2Fplatform.blossom-kc.com%2Fwawiwa_tech%2...

14/11/24, 9:20 p.m.

wawiwa

Componente de función - Conclusión

- Una función JavaScript/ES6
- Retorna un elemento React (JSX)
- Empieza con mayúscula (convenio de denominación)
- Toma props como parámetro si es necesario

PDF.js viewer

wawiwa

Componente de Clase

El segundo tipo es de el componente de clase.

Usamos una clase ES6 para definir un componente y retornar JSX.

Un ejemplo de componente de clase:

```
class Welcome extends React.Component {
  render() {
    return <h1>Hello, {this.props.name}</h1>;
  }
}
```

Diferente a los componentes funcionales, los componentes de clase deben tener un método render() adicional para retornar JSX.

PDF.js viewer

wawiwa

¿Por qué tenemos Componentes de Clase?

Solíamos usar componentes de clases por el “state”. En versiones más antiguas de React (versión < 16.8), no era posible usar estados dentro de los componentes funcionales.

Como resultado, necesitábamos los componentes funcionales para presentar el UI, y los componentes de clase manejaban los datos y otras actividades (como las funciones del ciclo de vida).

Con la llegada de React Hooks, ahora podemos usar estados en los componentes funcionales.

Los dos tipos de componentes son equivalentes desde el punto de vista de React

En este curso, usaremos únicamente componentes de función.

* state - una variable que es parte del componente.

PDF.js viewer

https://platform.blossom-kc.com/wawiwa_tech/Launcher?assignment=38867&ref=https%3A%2F%2Fplatform.blossom-kc.com%2Fwawiwa_tech%2...

14/11/24, 9:20 p.m.

wawiwa

Componente de clase - Conclusión

- Una clase ES6, será un componente una vez que 'extiende' un componente React.
- Tome Props (en el constructor) si es necesario
- Tiene un método render( ) para retornar JSX

PDF.js viewer

wawiwa

Props del componente de React

Props

Un concepto importante de los componentes.

Los props de React **transportan información de un componente a otro**.

Los props transportan información solo en una dirección (**únicamente del componente padre a los hijos**). No es posible usar props para transferir información del componente hijo al padre o componentes del mismo nivel.

Vale la pena señalar que la información del prop no es restringida a variables, también se pueden pasar funciones y objetos.
Los props se declaran de las siguientes maneras:

1. `const name = props.name;`
2. `function RandomName({name}) {...}`

https://platform.blossom-kc.com/wawiwa_tech/Launcher?assignment=38867&ref=https%3A%2F%2Fplatform.blossom-kc.com%2Fwawiwa_tech%2...

14/11/24, 9:20 p.m.

wawiwa

Props del componente de React - Ejemplos de Declaración

```
import React from 'react';
import Button from './Button';

export default function App() {

  return (
    <>
      <Button label="Click me!" text="Button clicked!"/>
    </>
  );
}
```


```
import React from 'react';

function Button(props) {

    function handleClick() {
        alert(props.text);
    }

  return (
    <button onClick={handleClick}>
      {props.label}
    </button>
  );
}

export default Button;
```

```
import React from 'react';

function Button({text, label}) {

    function handleClick() {
        alert(text);
    }

  return (
    <button onClick={handleClick}>
      {label}
    </button>
  );
}

export default Button;
```


PDF.js viewer

16/18

https://platform.blossom-kc.com/wawiwa_tech/Launcher?assignment=38867&ref=https%3A%2F%2Fplatform.blossom-kc.com%2Fwawiwa_tech%2...

wawiwa

Ejercicio 2

Crea un componente funcional que sea responsable del ticket de un pasajero.

Los componentes deben recibir estos parámetros:

- Name
- Destination
- Gender (Mr/Mrs)
- Seat

El componente va a retornar un div con estos detalles.

PDF.js viewer

https://platform.blossom-kc.com/wawiwa_tech/Launcher?assignment=38867&ref=https%3A%2F%2Fplatform.blossom-kc.com%2Fwawiwa_tech%2...

wawiwa

Ejercicio 3

Mueve el componente funcional del ejercicio 2 a un nuevo archivo.

Renderiza los tickets otra vez en la pantalla.

For example:

Ticket Details

Name: John Doe

Destination: New York

Gender: Mr

Seat: 14A

Ticket Details

Name: Alex Doe

Destination: Tel Aviv

Gender: Mr

Seat: 17A

PDF.js viewer