

React.JS

Introducción

Módulo 02

Introducción JSX

Introducción JSX

```
export class Nav extends React.Component {  
  constructor(props) {  
    super(props);  
  }  
  
  render() {  
    <nav>  
      <Logo></Logo>  
      <ul>  
        <li><a href="#">Item</a></li>  
      </ul>  
    </nav>  
  }  
}
```

Observa el código anterior. Seguro habrás notado una sintaxis particular: HTML dentro de JavaScript. ¿Qué es eso?

Se llama **JSX** y es una extensión de la sintaxis de JavaScript específica de React.

¿Cómo se crean los elementos?

React.js se basa en componentes, y vimos que un componente es, a grandes rasgos, una pieza de código que recibe unas props y retorna un Elemento React.

¿Cómo se crean los elementos React? Se crean a través de la función ***React.createElement()***. Esta función recibe tres parámetros: la etiqueta HTML que lo define, las props que se le pasan y los elementos hijos que tiene.

A través de *React.createElement()* podemos crear cualquier elemento de React.

Veamos el ejemplo anterior, pero utilizando `React.createElement`

```
export class Nav extends React.Component {  
  
  constructor(props) {  
    super(props);  
  }  
  
  render() {  
    return React.createElement('nav', {},  
      React.createElement('Logo', {}, null),  
      React.createElement('ul', {},  
        React.createElement('li', {},  
          React.createElement('a', {href: '#'}, 'Item')  
        )  
      )  
    );  
  }  
}
```



Excelente. Vamos a hacer una aplicación entera con eso...
Imaginemos hacer todo un sistema de administración basándonos en *React.createElement*.

Crear los elementos con *React.createElement()* es excelente pero poco “amigable” cuando uno está desarrollando. Es por eso que, en React.js, existe JSX:

JSX es una sintaxis más amigable para la función *React.createElement()*

El mismo ejemplo de antes, pero aplicando JSX, se escribiría así:

```
export class Nav extends React.Component {  
  
  constructor(props) {  
    super(props);  
  }  
  
  render() {  
    <nav>  
      <Logo></Logo>  
      <ul>  
        <li><a href="#">Item</a></li>  
      </ul>  
    </nav>  
  }  
}
```



Sin embargo, recordemos que esto es solamente una sintaxis más amigable.

Mediante el transpilador Babel, React interpreta el código JSX y lo transforma en llamadas a la función *React.createElement()*



- JSX acepta cualquier expresión JavaScript como contenido

```
<div>{this.state.mensaje}</div>
```

- JSX se puede usar como expresión

```
const el = <div>Hola</div>
```

- JSX permite crear elementos HTML y extenderlos

```
<div tabIndex="0" className='boton'>  
  <span>  
    Click aqui  
  </span>  
</div>
```

Todo elemento JSX se convierte luego en llamadas a *React.createElement()* y se procesan el resultado de esas llamadas.

JSX es una sintaxis amigable para dicha función.

¡Muchas gracias!

¡Sigamos trabajando!