

# Bootcamp Front End, ReactJs:

## Lectura 6

Ing. Pedro Rotta

Universidad de Piura - Vida Universitaria

Febrero-2022

# React

React es una librería de JavaScript para crear interfaces gráficas de usuario. Es muy usado para crear **aplicaciones de una página**. Permite crear múltiples proyectos rehusando código y **componentes**.

React fue creado por FB para poder hacer un código más limpio, sencillo y fácil de manejar. Debido a su facilidad de uso, React permite que múltiples desarrolladores creen sus propios componentes y se integren en una sola aplicación.

Entre sus ventajas están, **rehusabilidad, facilidad y velocidad**.

# Conceptos básicos

**El CDN** de react te permite una aproximación para practicar y cultivar tu código como desarrollador, para usarlo podemos ir a la página oficial de React:

`https:  
//es.reactjs.org/docs/add-react-to-a-website.html.`

# La función Render

La función principal de react, es tener varias formas de renderizar una página web. La función que se utiliza en este proceso es `ReactDOM.render()`.

La función `ReactDOM.render()` tiene 2 atributos, el código en html y el valor del objeto raíz. Su sintaxis sería:

```
ReactDOM.render(<h1>Hola mundo</h1>),  
document.getElementById("root");
```

Puedes crear tus propias etiquetas.

# React JSX

**JSX** es el estándar para escribir etiquetas XML en JavaScript. Permite usar html en javascript de manera directa y su sintaxis es muy parecida a html, por lo que es muy fácil de adaptar.

Se puede elegir entre hacerlo directo o guardar parte del contenido html en una variable antes de guardarlo en el div principal.

```
const titulo = <h1>Me encanta JavaScript</h1>;  
ReactDOM.render(titulo,  
document.getElementById("root"));
```

También puedes añadir directamente expresiones a tu código original.

```
const titulo = <h1>Me encanta JavaScript {5+5} veces  
más</h1>;  
ReactDOM.render(titulo,  
document.getElementById("root"));
```

# Elementos sin salida

Para definir imágenes y elementos sin marca de salida, podemos incluirlos usando un cierre, dentro de la misma marca de entrada:

```
const myelement = <input type="text" />;
```

Además el atributo class, pasa a ser el atributo **className**

Para usar condicionales, podemos usar condicionales, pero con la siguiente sintaxis:

```
const x = 5;  
let text = "Goodbye";  
if (x < 10) {  
  text = "Hello";  
}  
const myelement = <h1>text</h1>;
```

# Components

**Componentes** en React son extractos de funciones que retornan objetos de HTML en el DOM.

La sintaxis para crear un componente es :

```
class Car extends React.Component {  
  render() {  
    return <h2>Hi, I am a Car!</h2>;  
  }  
}
```

# Componentes

Podemos crear un componente a través de una función:

```
function Car() {  
  return <h2>Hi, I am a Car!</h2>;  
}
```

Para renderizar un componente, escribimos:

```
ReactDOM.render(<Car />,  
document.getElementById('root'));
```