



Toma de contacto ReactJS

Introducción

ReactJS es una librería de **JavaScript** enfocada a construir interfaces de usuario interactivas. Esta librería está basada en **Componentes** reutilizables que manejan su propio estado con los que poder pasar datos a través de la aplicación.

Estos componentes los crearemos con una sintaxis similar a XML llamada **JSX**, la cual es bastante útil como ayuda visual a la hora de trabajar unificando diversas tecnologías.

Este es un ejemplo de como JSX combina varios de los lenguajes aprendidos anteriormente:

```
const title = <h1>Student</h1>;
```

Cómo crear una App de React

Para crear una nueva App de React necesitarás tener Node ≥ 10.16 y Npm ≥ 5.6 instalados en tu máquina y ejecutar el siguiente código:

```
npm create vite@latest //Creamos un proyecto con Vite
//Indicamos el nombre del proyecto
//Elegimos react y react
//Seguimos las instrucciones para instalar las dependencias y arrancar el proyecto
```

Estructura de un proyecto de React

Vamos a ver qué ficheros trae definidos un proyecto de React y para qué sirve cada uno de ellos.

Carpeta public: En esta carpeta se encuentran los assets públicos de nuestra aplicación, tales como el `favicon`, `manifest` o `robots`.

index.html: Dentro de este fichero nos encontramos un `<div id="root"></div>` que constituye nuestro punto de entrada, es decir dentro de dicho div va a correr nuestra aplicación de React.

React necesita de una librería para interactuar con el navegador, dicha librería se denomina react-dom. Por ello si pasamos a la carpeta src podemos ver que la estamos importando.

Carpeta src: En esta carpeta irá toda la estructura de la aplicación, empezando por el archivo `main.jsx`:

```
import React from 'react';
import ReactDOM from 'react-dom';
import App from './App';

ReactDOM.render(
  <React.StrictMode>
    <App />
  </React.StrictMode>,
  document.getElementById('root')
);
```

En este archivo podemos ver las siguientes importaciones:

- La importación de `React`, ya que todos los archivos JSX deberán contenerlo.
- `ReactDOM` se encargará de inyectar toda nuestra aplicación en el `index.html` tal y como hemos visto anteriormente.

- Al igual que en HTML tendremos que importar las `hojas de estilo` para poder usarlas, pero en este caso tendremos que hacerlo como si fueran un archivo más.
- `App` es el componente principal de la aplicación, del cual vamos a profundizar en la siguiente sección.

Componente App

Los cimientos donde irán los `componentes` de React están en `App.js`, el cual es conveniente convertir en `.jsx` para un buen uso de la sintaxis anteriormente mencionada.

```
import React from 'react';
import './App.scss';

function App() {
  return (
    <div className="App">
      <p>Lorem Ipsum</p>
    </div>
  );
}

export default App;
```

Lo que este componente retorne será lo que veamos en nuestra aplicación al arrancarla. Este componente, al igual que el resto, puede contener tanto etiquetas HTML como otros componentes en forma de etiquetas.