

¿Qué es JSX?

JSX

Es una extensión del lenguaje desarrollada por Facebook para React.

Parece JS pero muestra código de HTML, y básicamente es un lenguaje de Template que muestra el HTML pero tiene todas las funciones de JavaScript.

Una vez compilado son archivos JS con funciones y objetos.

```
Const nombre = "Yoni Escobar"  
Const total = 150 + 350
```

```
<h1> Bienvenido mi nombre es {nombre} y total es {total}</h1>
```

Regla

JSX

A diferencia de HTML, que no es estricto, en JSX si un elemento HTML tiene una etiqueta de apertura, deberás tener también la de cierre.

Las etiquetas de solo apertura como `<link>` `` o `<input>` deberán finalizar con `/>`

Cada componente debe un return

En este return debe haber máximo un solo elemento en el nivel máximo

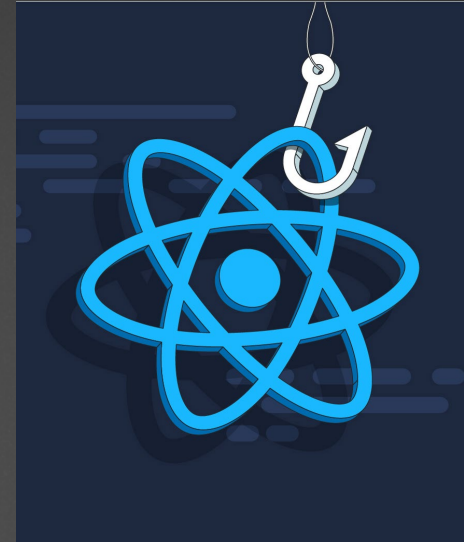
¿Qué son los Hooks?

React Hooks o Hooks?

React cuenta con una API muy sencilla que te permite crear todo tipo de aplicaciones por medio de algo llamado Hooks.

Los Hooks están disponibles desde la versión 16.8, previo a ello se tenían que crear classes para crear y modificar el state, con los Hooks no es necesario

Los Hooks se dividen en básicos y Adicionales



Básicos

- useState
- useEffect
- useContext

Adicionales

- useReducer
- useCallback
- useMemo
- useRef
- useImperativeHandle
- useEffect
- useDebugValue

<https://youtu.be/LlvBzyy-558>

Crear tus propios Hooks

También es posible crear tus propios Hooks, de esta forma podrás separar tu código en funciones reutilizables y sacar todo el beneficio de lo que React ofrece

State – Estado en React

El State o Estado es básicamente eso; cuál es el estado de nuestra aplicación.

El Estado es una variable con información relevante en nuestra aplicación de React, algunas veces el state pertenece a un componente en específico o algunas veces deseas compartirlo a lo largo de diferentes componentes.

El state es creado con la función `useState()`;

```
import { useState } from "react"

const [cliente, setCliente] = useState({});
const [total, setTotal] = useState(0);
const [clientes, setClientes] = useState([]);
const [modal, setModal] = useState(false);
```

React reacciona en base al State
Cada vez el state cambia, la aplicación va a renderizar y actualizarse con esos cambios

Reglas de los Hooks

Los Hooks se colocan en la parte superior de tus componentes de React.

No se deben colocar dentro de condicionales, tampoco después de un return.

UseEffect

useEffect siempre es un callback, que se ejecuta cuando un state cambia o cuando el componente esta listo.

Al ejecutarse automaticamente cuando el componente esta listo, es un excelente lugar para colocar código para consultar una API o LocalStorage

Debido a que le podemos pasar una dependencia y estar escuchando por los cambios que sucedan en una variable, puede actualizar el componente cuando ese cambio suceda

```
import { useEffect } from "react"

useEffect( () => {
  console.log('El Componente esta listo');
}, []);
```

- Crear un menú
- Crear un formulario
- consumir api

- Vscode y las extensiones
- nvm list
- Node.js version 18.20

Pasos para resolver el ejercicio planteado.

1. `npx create-react-app menu-opciones`
2. Limpiar el proyecto (quitar archivos que no se van a utilizar)
3. Instalar Librería

```
npm i react-router-dom  
npm i axios  
npm i bootstrap  
npm i sweetalert2-react-content  
npm i @fortawesome/fontawesome-free
```

Instalar librerías todo junto:

```
npm i react-router-dom axios bootstrap sweetalert2 sweetalert2-react-content  
@fortawesome/fontawesome-free
```