**¿Qué es JSX?**

JSX significa JavaScript XML. JSX nos permite escribir HTML en React. JSX facilita escribir y agregar HTML en React.

EJEMPLO 1: **React con condicionales/string/boleanos:**

import React from "react";

import ReactDOM from "react-dom/client";

const root = ReactDOM.createRoot(document.getElementById("root"));

function Saludar() {

  const married = true;

  return (

    <div>

      <h1>{married ? "estoy casado" : "no estoy casado"}</h1>

      <p>Esto es un parrafo</p>

    </div>

  );

}

//ESCRIBIMOS LO QUE CONTIENE ESA FUNCION, LLAMANDO A LA MISMA FUNCION. De paso lo reutilizamos.

root.render(

  <div>

    <Saludar></Saludar>

    <Saludar></Saludar>

  </div>

);

EJEMPLO 2: **React pasando un objeto a un JSON**

import React from "react";

import ReactDOM from "react-dom/client";

const root = ReactDOM.createRoot(document.getElementById("root"));

function Saludar() {

  const user = {

    nombre: "juan",

    edad: 24,

    dni: 12345678,

  };

  //AHORA RETORNAMOS ESE OBJETO (QUE SE ENCUENTRA DENTRO DE LA FUNCION) CONVERTIDO A JSON.

  return (

    <div>

      <h1>{JSON.stringify(user)}</h1>

    </div>

  );

}

//ESCRIBIMOS LO QUE CONTIENE ESA FUNCION, LLAMANDO A LA MISMA FUNCION. De paso lo reutilizamos.

root.render(

  <div>

    <Saludar></Saludar>

    <Saludar></Saludar>

  </div>

);

EJEMPLO 3: **React pasando un objeto a elementos HTML dentro de un COMPONENTE COMO VENIMOS HACIENDO CON TODO.**

import React from "react";

import ReactDOM from "react-dom/client";

const root = ReactDOM.createRoot(document.getElementById("root"));

function Saludar() {

  const user = {

    nombre: "juan",

    edad: 24,

    dni: 12345678,

  };

  //AHORA RETORNAMOS ESE OBJETO (QUE SE ENCUENTRA DENTRO DE LA FUNCION) CONVERTIDO A JSON.

  return (

    <div>

      <h1>{user.nombre}</h1>

      <h1>{user.edad}</h1>

      <h1>{user.dni}</h1>

    </div>

  );

}

//ESCRIBIMOS LO QUE CONTIENE ESA FUNCION, LLAMANDO A LA MISMA FUNCION. De paso lo reutilizamos.

root.render(

  <div>

    <Saludar></Saludar>

    <Saludar></Saludar>

  </div>

);

EJEMPLO 4: **React viendo lo que hay dentro de una constante, pasándola a string.**

import React from "react";

import ReactDOM from "react-dom/client";

const root = ReactDOM.createRoot(document.getElementById("root"));

function Saludar() {

  const married = true;

  return (

    <div>

      <h1>{married.toString()}</h1>

    </div>

  );

}

//ESCRIBIMOS LO QUE CONTIENE ESA FUNCION, LLAMANDO A LA MISMA FUNCION. De paso lo reutilizamos.

root.render(

  <div>

    <Saludar></Saludar>

    <Saludar></Saludar>

  </div>

);

EJEMPLO 5: **React operaciones matemáticas con funciones.**

import React from "react";

import ReactDOM from "react-dom/client";

const root = ReactDOM.createRoot(document.getElementById("root"));

function Saludar() {

  function add(x,y){

    return x + y;

  }

  return (

    <div>

      <h1>{add(50,50)}</h1>

    </div>

  );

}

//ESCRIBIMOS LO QUE CONTIENE ESA FUNCION, LLAMANDO A LA MISMA FUNCION. De paso lo reutilizamos.

root.render(

  <div>

    <Saludar></Saludar>

    <Saludar></Saludar>

  </div>

);