**Chapter 04 - Talk is cheap, show me the code!**

**Is JSX mandatory for React?**

JSX is not mandatory for React. It is a JavaScript extension that allows you to write HTML in your JavaScript code. JSX makes it easier to write React code, but it is not required.

If you do not want to use JSX, you can write your React code in plain JavaScript. However, using JSX can make your code more readable and easier to maintain.

Here is an example of a React component written with JSX:

const MyComponent = () => {

return (

<div>

<h1>Hello, world!</h1>

</div>

);

};

This component renders a div element with the text "Hello, world!" inside of it.

Here is the same component written in plain JavaScript:

const MyComponent = () => {

const element = React.createElement(

'div',

null,

React.createElement('h1', null, 'Hello, world!')

);

return element;

};

This component renders the same div element as the previous example, but it is written in plain JavaScript instead of JSX.

As you can see, JSX can make your React code more readable and easier to write. However, it is not required to use JSX with React.

<https://legacy.reactjs.org/docs/react-without-jsx.html>

**Is ES6 mandatory for React?**

No, ES6 is not mandatory for React. However, it is highly recommended, as most modern React projects use ES6 features. ES6, also known as ECMAScript 2015, is a significant update to the JavaScript language that includes many new features, such as classes, arrow functions, and template literals. These features can make your code more concise, readable, and maintainable.

**Here are some of the benefits of using ES6 in React:**

* Conciseness:

ES6 features such as arrow functions and template literals can make your code more concise and easier to read.

* Readability:

ES6 features such as classes and modules can make your code more organized and easier to understand.

* Maintainability:

ES6 features such as const and let can help you write more maintainable code.

**What is and <React. Fragment></React.Fragment> and <></> ?**

React Fragments allow you to wrap multiple elements without adding an extra DOM element. In our case, we replace the <div> warpper with the Fragment element in the <Column /> component as follows:

function Column () {

return (

<React.Fragment>

<td>Hello World!</ td>

<td>Hello Educative!</ td>

</React.Fragment>

); }

<></> this is shorthand syntax for React Fragments

there is a very important difference between React fragment and Empty tag, that is, one can use the key attribute in React.Fragment syntaxes but in empty tag syntax that’s not possible.

[**https://medium.com/fasal-engineering/what-are-react-fragments-or-the-react-empty-tags-190253582905**](https://medium.com/fasal-engineering/what-are-react-fragments-or-the-react-empty-tags-190253582905)