What is JSX?

Is like HTML syntax but not exact HTML. Tag syntax is neither a string nor HTML. It’s syntax extension which has the full power of JavaScript.

Superpower of JSX

const element = <h1>Hello, world!</h1>;

The Above code is much easy than the below mention code

React.createElement(

'div',

{className: 'sidebar'},

‘HELLO Word’

)

JSX just provides syntactic sugar for the React.createElement(component, props, ...children) function.

You can easily write…

Const bookName = ‘Wings of Fire’

const Title3 = () => (  /// this is react functional component

    <div>

        <p>This is title three</p>

<p>{bookName}</p>

<p>{ 5 \* 8}</p>

    </div>

)

**Role of type attribute in script tag? What options can I use there?**

The type attribute specifies the type of the script.

The type attribute identifies the content between the <script> and </script> tags

text/javascript : It is the basic standard of writing javascript code inside the <script> tag.

### Syntax

<script type="text/javascript"></script>

text/ecmascript : this value indicates that the script is following the EcmaScript standards.

module: This value tells the browser that the script is a module that can import or export other files or modules inside it.

text/babel : This value indicates that the script is a babel type and required bable to transpile it.

text/typescript: As the name suggest the script is written in TypeScript.

## {TitleComponent} vs {<TitleComponent/>} vs {<TitleComponent></TitleComponent>} in JSX.

A: The Difference is stated below:

* {TitleComponent}: This value describes the TitleComponent as a javascript expression or a variable. The {} can embed a javascript expression or a variable inside it.
* <TitleComponent/> : This value represents a Component that is basically returning Some JSX value. In simple terms TitleComponent a function that is returning a JSX value. A component is written inside the {< />} expression.
* <TitleComponent></TitleComponent> : <TitleComponent /> and <TitleComponent></TitleComponent> are equivalent only when < TitleComponent /> has no child components. The opening and closing tags are created to include the child components.

### Example

<TitleComponent>

<FirstChildComponent />

<SecondChildComponent />

<ThirdChildComponent />

</TitleComponent>