**Chapter 03: Laying The Foundaton**

***Theory:***

Q) What is JSX

JSX stands for JavaScript XML. It is a “Syntax Extension to JavaScript” allows you to write HTML-like code within our JavaScript code.

NOTE 1: We can build an APP without using JSX. BUT it makes the developers life very easy by allowing us to write HTML-like code with in the JS code. Developing an APP using JSX will be easier and faster.

NOTE 2: JSX is NOT a part of React JS. It exists with in JS code. That’s why its called Syntax Extension for JS.

Q) Super Powers of JSX

1. Using JSX, we can write HTML-like content inside Javascript.

It provides a superpower to write logic and markup of a component inside a single .js file. JSX is easy to maintain and debug.

1. Developing an APP using JSX will be easier and faster

Q) What is the Role of “type” attribute in the script tag? What options can be used there?

The "type" attribute in the <script> tag specifies the type of content that is contained within the script block OR the file that is specified in "src" attribute.

The "type" attribute is important because it helps the browser determine whether the content should be treated as JavaScript or another programming language.

SOME common options for “type” attribute are

1. <script type=”text/javascript”></script>

In HTML5, the "type" attribute is optional. Browsers assume "text/javascript" by default if the "type" attribute is omitted.

1. <script type="module">

// JavaScript module code here

</script>

The "type" attribute is set to "module" tells the browser that the script is a module that can import or export other files or modules inside it.

1. <script type="application/json">

{

"key": "value",

"array": [1, 2, 3]

}

</script>

The "type" attribute can be used to embed JSON data within a script tag

1. <script type="text/python">

# Python code here

</script>

We can specify the "type" attribute for other scripting and programming languages.

For example, if we're embedding code for Python or Ruby, you would specify the appropriate content type as type="text/python"

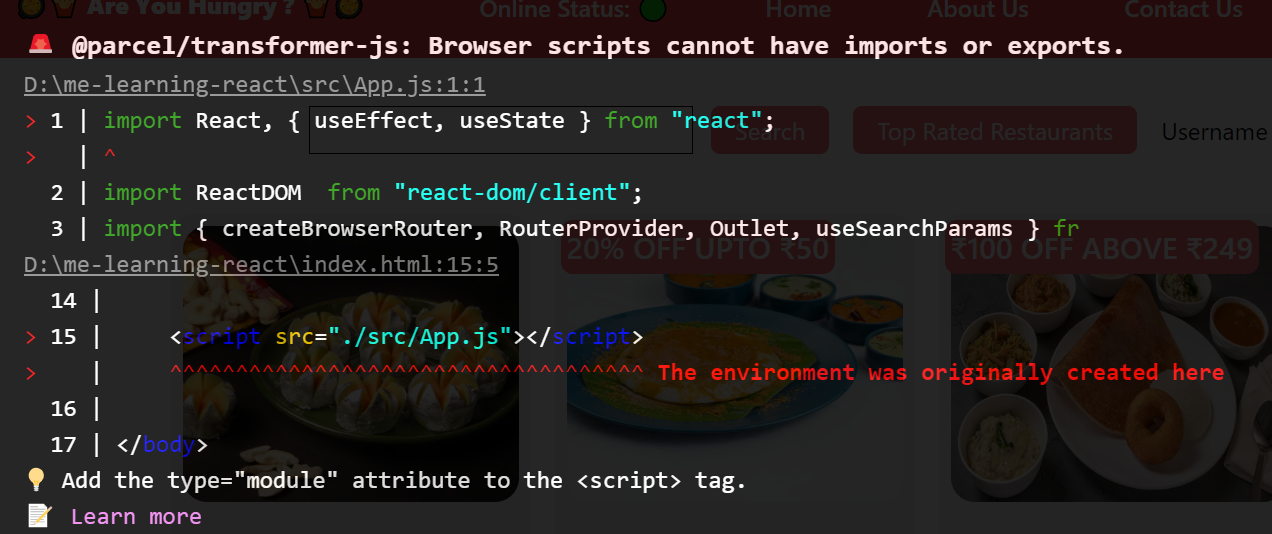
**NOTE 1:** Legacy and Deprecated Types: In the past, we might encounter scripts with "type" attributes like "text/vbscript" for VBScript or "text/php" for PHP. These are largely deprecated and not recommended for modern web development.

**NOTE 2:** In modern web development, it's common to use "text/javascript" or "module" for JavaScript code and omit the "type" attribute when using JavaScript. Browsers will correctly interpret the content as JavaScript by default.

BUT If you're working with ECMAScript modules or embedding non-JavaScript data in a script tag, specifying the appropriate "type" attribute is important for proper interpretation and execution.

**NOTE 3:** When I removed the type=”module” from my app’s index.html page, it gave me the following error

This is because when we did not give the type attribute, the browser treats the script as normal java script file, which can not have the imports and esports. That’s why its IMPORTANT to mention the appropriate value when the content written are not plane JavaScript.



Q) {TitleComponent} VS <TitleComponent /> VS

<TitleComponent > <TitleComponent /> in JSX

1. **{TitleComponent}:** This value describes the TitleComponent as a javascript expression or a variable. The {} can embed a javascript expression or a variable inside it.
2. **{<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.
3. **<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.

Eg:

<TitleComponent>

<FirstChildComponent />

<SecondChildComponent />

</TitleComponent>