

## Chapter 03 - Laying The Foundation

### Q.1&2 What is JSX? What are the superpowers of JSX?

**Ans:** JSX is a syntax extension for Javascript that lets you write HTML-Like markup inside a JS file.

**Ex:** `const element = <h1>Welcome to Namaste React!</h1>;`

#### Characteristics:

- JSX is not mandatory to use. There are other ways to achieve the same thing but using JSX makes it easier to develop react applications.
- **JSX allows writing expressions in { }. The expression can be any JS expression or React variable.**
- To insert a large block of HTML we have to write it in a parenthesis i.e, ().
- JSX produces react elements.
- JSX follows the XML rule.
- After compilation, JSX expressions become regular JavaScript function calls.
- JSX uses camelcase notation for naming HTML attributes. For example, tabIndex in HTML is used as tabIndex in JSX.

#### Advantages:

- JSX makes it easier to write or add HTML in React.
- JSX can easily convert HTML tags to react elements.
- It is faster than regular JavaScript.
- JSX allows us to put HTML elements in DOM without using [appendChild\(\)](#) or [createElement\(\)](#) method.
- As JSX is an expression, we can use it inside of if statements and for loops, assign it to variables, accept it as arguments, or return it from functions.
- **JSX prevents XSS (cross-site-scripting) attacks popularly known as injection attacks.**
- It is type-safe, and most of the errors can be found at compilation time.

#### Disadvantages:

- JSX throws an error if the HTML is not correct.
- In JSX HTML code must be wrapped in one top-level element otherwise it will give an error.
- If HTML elements are not properly closed JSX will give an error.

### Q.3 What is the role of type attribute in <script> tag? What options can I use there?

**Ans:** The HTML <script> type attribute is used to tell the specific MIME type of script and specify the content of the tag. It has a default value which is "text/javascript".

**Syntax:** `<script type="media_type">`

Here media\_type can take up -

- text/javascript
- text/ecmascript
- application/ecmascript
- application/javascript

## Chapter 03 - Laying The Foundation

**Q.4** `{TitleComponent}` vs `{<TitleComponent/>}` vs `{<TitleComponent></TitleComponent>}` in JSX.

**Ans:** `{TitleComponent}` is a react element.

`{<TitleComponent/>}` is a react component.

`{<TitleComponent></TitleComponent>}` is also a react component. This and above i.e., second one are same.