Chapter 03 Assignment Theory

1. What is JSX?
   JSX is an extension of the JavaScript language which provides a way to structure component rendering using syntax similar to HTML. JSX gives us the ability to write HTML elements in Javascript and place them in the DOM by converting the HTML tags into React elements without the need for other methods like createElement() or appendChild(). This combination of Javascript and HTML leads to having more powerful applications with boosted performance.

2. Superpowers of JSX
   a. It makes it easy to write HTML in JS.
   b. JSX elements can act as values for identifiers. Example: *const hello = <h1>Hello, World!</h1>;*
   c. JSX supports attributes the same way as HTML does, apart from the fact that JSX uses camelCase.
   d. JSX supports all JavaScript Expressions by wrapping them inside a pair of curly brackets.
   e. It prevents cross site scripting attacks popularly known as injection attacks.

3. Role of type attribute in script tag? What option can I use there ?
   This attribute indicates the type of script represented. If it is not set, it indicates that the script is a "classic script", containing JS code. Other values it can take are:
   Module: This value causes the code to be treated as a JS module.
   Importmap: This indicates that the body of the element contains an import map. Import map is a JSON object that can be used to control how the browser resolves module specifiers when importing JS modules

4. {TitleComponent}vs{<TitleComponent/>}vs{<TitleComponent></TitleComponent>} in JSX.
   {TitleComponent} – It a JSX expression/variable
   {<TitleComponent/>} - It is a component without any props
   {<TitleComponent></TitleComponent>} – It is similar to an HTML element, and we can include other JS code or embed and execute other components within TitleComponent