Assignment 2

Rojin Bijukchhe

1. How does JSX differ from HTML?

JSX(JavaScript Syntax Extension) is a syntax extension for JavaScript.
Commonly used in React to Describe the

- Structure
  Static component
- Elements
  Dynamic component

Whereas HTML is Markup Language and is directly understood by browsers.
Commonly used for

Structure
- Static content defination.

Feature	JSX	HTML
Type	JavaScript Syntax Extension	Markup Language
Functionality	Can include code and expressions	Static content definition
Use Case	Building UI components in React	Creating web page structure
Compilation	Transformed to JavaScript functions	Directly interpreted by browser
Elements	Can be combined into reusable components	Standalone elements

Difference between JSX and HTML

2. Why is JSX used in React?

React applications can be written without JSX, but there are several advantages to using it:

- Readability: JSX offers a familiar HTML-like syntax for describing UI components.

- Component Composition: JSX allows you to write component structures that closely resemble the final UI layout.
- Combines mark ups and logic: JSX combines markup and logic within a component, it keeps these concerns separate from the HTML in your main web page.

3. Can you embed JavaScript expressions in JSX? If so, how?

- Yes, you can definitely embed JavaScript expressions in JSX.
  - Curly Braces: By using Curly Braces {} to enclose any valid JS expression within our JSX code.
  - Expression Evaluation: The expression inside the curly braces gets evaluated during the rendering process. The result of the evaluation is then inserted into the JSX element.

```
function App3() {
  /**
   * Variables: You can directly use variables
   *  declared outside the JSX code:
   */
  const name = "World";
  const element = <h1>Hello, {name}!</h1>;
  return <div class="result">{element}</div>;
}
```

```
/**
 * Expressions: Any valid JavaScript expression can be used:
 */
const age = 30;
const message = <p>You are {age} years old.</p>;
```

```
/**
 * Function Calls: You can call functions and
 * include the result in the JSX:
 */

function formatName(user) {
  return user.firstName + " " + user.lastName;
}

const user = { firstName: "Rojin", lastName: "Bijukchhe" };
const greeting = <h2>Welcome, {formatName(user)}</h2>;
```

4. How do you write comments in JSX?

{/* Comment Here */}

5. Explain the significance of curly braces {} in JSX.

Curly braces **{ }** play a critical role in JSX by enabling us to embed **JavaScript expressions and logic** within our component markup (HTML file).

6. Can JSX be directly rendered to the DOM?

No, JSX code cannot be directly rendered to the DOM. It needs to be parsed by a parsing tool (BABEL) before it can run in the browser.

7. What is the purpose of Babel in relation to JSX?

Purpose (Workflow) for using JSX:

- We write our UI components using JSX within our React code.
- During the build process, tools like **Babel** transpile our JSX code into regular JavaScript function calls.
- The resulting JavaScript code creates React elements representing the UI.
- React manages a virtual DOM and efficiently updates the actual browser DOM based on changes.