

HOMEWORK - 2

// 1. How does JSX differ from HTML?

Ans:

- JSX is a syntax extension for JavaScript, while HTML is a markup language.
- JSX allows to include expressions and function within the syntax, while HTML only allows for static text.
- JSX is typically used in React applications, while HTML is used to create web pages.

// 2. Why is JSX used in React?

Ans:

- JSX allows us to write HTML elements in JavaScript and place them in the DOM without any createElement() and/or appendChild() methods.

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

Ans:

Yes, with JSX expressions can be written in curly braces {}. The expression can be a React variable, or property, or any other valid JavaScript expression. JSX will execute the expression and return the result.

e.g.1 to execute the expression `5 + 5`

`const myElement = <h1> React is {5 + 5} times better with JSX </h1>`

result: React is 10 times better with JSX

e.g.2 `const name = 'Belay Work'`

`const element = <h1> Hello, {name} </h1>`

result: Hello Belay Work

// 4. How do you write comments in JSX?

Ans:

Comments may sometimes be needed for the readability of our code. JSX has its own way of putting comments. The syntax `{/* */}` should be used to wrap around the comment text.

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

Ans:

In JSX, curly braces {} are special syntax. It is used to evaluate a JavaScript expression during compilation. The expression can be a

variable, function calls, an object, or any other JavaScript code that resolve into a value

// 6. Can JSX be directly rendered to the DOM?

Ans:

No, JSX cannot be directly rendered to the DOM. In order JSX to be rendered in the browser JSX needs to be transpiled into a regular JavaScript using a tool like Babel. React provides a virtual DOM that efficiently update the actual DOM based on changes to the JSX elements. So, while JSX itself cannot be directly rendered to the DOM, it is an integral part of many modern web development workflows, particularly those using React or similar libraries.

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

Ans:

We use Babel with React to transpile the JSX code into simple React functions that can be understood by browsers. Using this way we can assure that our JSX code can work in almost any browser.