# OBJECTIVES

## 1. Define JSX

* JSX (JavaScript XML) is a syntax extension for JavaScript used in React. It allows writing HTML-like code inside JavaScript, making UI code more readable and easier to write. JSX gets transpiled into React.createElement() calls by tools like Babel.  
    
  Example:  
  const element = <h1>Hello, JSX!</h1>;

## 2. Explain about ECMA Script

* ECMAScript (ES) is the official standard for JavaScript, maintained by ECMA International. It defines the rules, syntax, and features of the language. Versions are named ES5, ES6 (ES2015), ES7, ES8, etc. ES6 introduced major features like let, const, arrow functions, classes, template literals, and destructuring.

## 3. Explain React.createElement()

* React.createElement() is a core React method to create React elements without JSX.  
  Syntax: React.createElement(type, props, ...children)  
  Parameters:  
  - type: HTML tag or React component  
  - props: attributes or properties of the element  
  - children: nested elements or text  
  Example:  
  const element = React.createElement('h1', { className: 'title' }, 'Hello World');

## 4. Explain how to create React nodes with JSX

* A React node is anything React can render (elements, components, strings, numbers, etc.). With JSX, nodes are created by simply writing HTML-like syntax.  
  Example:  
  const node = <p>This is a paragraph</p>;

## 5. Define how to render JSX to DOM

* Use ReactDOM.createRoot() (React 18+) or ReactDOM.render() (React < 18).  
  Example:  
  import React from 'react';  
  import ReactDOM from 'react-dom/client';  
    
  const element = <h1>Hello JSX</h1>;  
  const root = ReactDOM.createRoot(document.getElementById('root'));  
  root.render(element);

## 6. Explain how to use JavaScript expressions in JSX

* Use curly braces { } to embed JavaScript expressions inside JSX.  
  Example:  
  const name = 'Ujjwal';  
  const element = <h1>Hello, {name}!</h1>;

## 7. Explain how to use inline CSS in JSX

* Use the style attribute with a JavaScript object. CSS property names must be camelCase.  
  Example:  
  const headingStyle = { color: 'blue', fontSize: '20px' };  
  const element = <h1 style={headingStyle}>Styled Heading</h1>;