# Module\_45\_React\_Core\_Concepts\_JSX\_Props\_State

#### **45-1 Module Introduction, Create React App, 6 Core Concepts**

* React also has six core concepts like JavaScript. Today’s module is dedicated to discussing these six-core things.

#### **45-2 Jsx, Dynamic Content, Dynamic Style In React**

* [JSX](https://www.geeksforgeeks.org/reactjs-introduction-jsx/) stands for JavaScript XML. It is simply a syntax extension of JavaScript. It allows us to directly write HTML in React (within JavaScript code). It is easy to create a template using JSX in React, but it is not a simple template language; instead, it comes with the full power of JavaScript. Learn More… [JSX Full Form](https://www.geeksforgeeks.org/jsx-full-form/)
* We will use **src/App.js** for HTML and JavaScript and **src/App.css** for CSS rules. Write JavaScript expression inside curly brace {} in HTML. We have to use **className** in HTML as the class attribute for avoiding conflict with the JavaScript **class** keyword.

#### **45-3 Create Component, Return Html From A Component**

* Delete all HTML elements from **App** function except main parent div with class name **‘App’**.
* Declare a function with capitalization, write any HTML code in it’s return ( ).
* Use that declared function name as an HTML tag into the main **App** function to execute that component.

#### **45-4 Pass Dynamic Data To Components, Props In React**

* If we add one or more **props** (attribute) in **components** (used as custom HTML tag), the corresponding function receives an argument (props) of an object that contains all the props as key-value pairs.

#### **45-5 (Advanced) Create Multiple Components From An Array Of Objects**

* We can create an array of objects and dynamically display them using the **map** method and **props**.

#### **45-6 Concept Recap, Jsx, Components, Props, Display Array Of Objects**

* We can use CSS both in **App.css** and inside the component (function) as an object and use it using **style**.

#### **45-7 (Advanced) State, Component State Hook And Set State Method**

* We have to *import* **useState** using **“import { useState } from 'react'”** for using it.
* **useState() -** return an array containing the passed value (number in this case) and a function. We can capture these two things by using de-structuring. i.e. **const [count, setCount] = useState(0);**
* We have to use **onClick=“functionName”** instead of **onclick=“functionName()”** for calling a function on click event in **react**.

#### **45-8 (Advanced) Load Dynamic Data, Api Call Useeffect Integrate State**

* **useEffect**() - is used for API call. It received two parameters, the first one is an anonymous function and the second one is the default value.
* **useState** and **useEffect** are totally new concepts. It will be more clear in the upcoming modules.

#### **45-9 Module Summary React Core Concepts Recap**

* The purpose of modules 44 & 45 was to get the primary concepts of the basic six things of React. They are **JSX, Components, props, event, state,** and Dynamic data load using **useEffect** and **useState.** Details discussion of these six concepts will come in the future module.