Introduction To The ReactJS

React is open-source JavaScript library for building user interfaces.

React is JavaScript library, not a framework.

React is focused on building user-interface, instead of navigations and API calls.

React has it's on virtual DOM.

React provides some inbuild classes and functions by using them we can create react elements, those react elements are used to create UI.

React is widely use for building single page applications (SAP).

Q) Why learn React?

- React is developed and maintain by Facebook.
- Huge community.
- In demand skillset.
- React has component (re-usable blocks) based architecture.
- React will efficiently handle rendering and updating of the UI components with its virtual DOM.
- DOM updates are handled gracefully in react.
- Once you get hands-on react, you can easily learn react-native to develop mobile application.

Q) What is single page application?

A single-page-application is a application which has only one HTML page. And the perfect example of single-page-applications are Netflix and Gmail.

In the single-page-application we can navigate around URL, and it seems like we are visiting different pages. But in the reality page never get changes only content of page changes. (It will re-write the page).

❖ **Advantages of SAP: -**
   1. High performance and stability.
   2. Code separation.
   3. Fast development.

Prerequisites: -

1. HTML
2. CSS
3. JavaScript fundamentals (ES6).

React Elements: -

The elements, which are created using React library are react elements.

To create react elements, react library provides us one pre-define function called **createElement( ).** Using this function we can create react elements.

 Ex.

    let element =React.createElement("h1",{id:"heading" }, "Welcome to ReactJS");

createElement( ) function have three parameter arguments as below: -

  1st argument: - Tag name for element which we want to create.

  2nd argument: - Object which contain attributes and their values for the element.

  3rd argument: - children/ innerHTML of an element.

ReactDOM: -

     ReactDOM is also known as virtual dom.

     ReactDOM is an important library and it is use to add react element in to the Actual DOM.

     ReactDOM has per-define function **render( ).**

EX.

     let element =React.createElement("h1",{id:"heading" ,width:'100px'}, "Welcome to ReactJS");

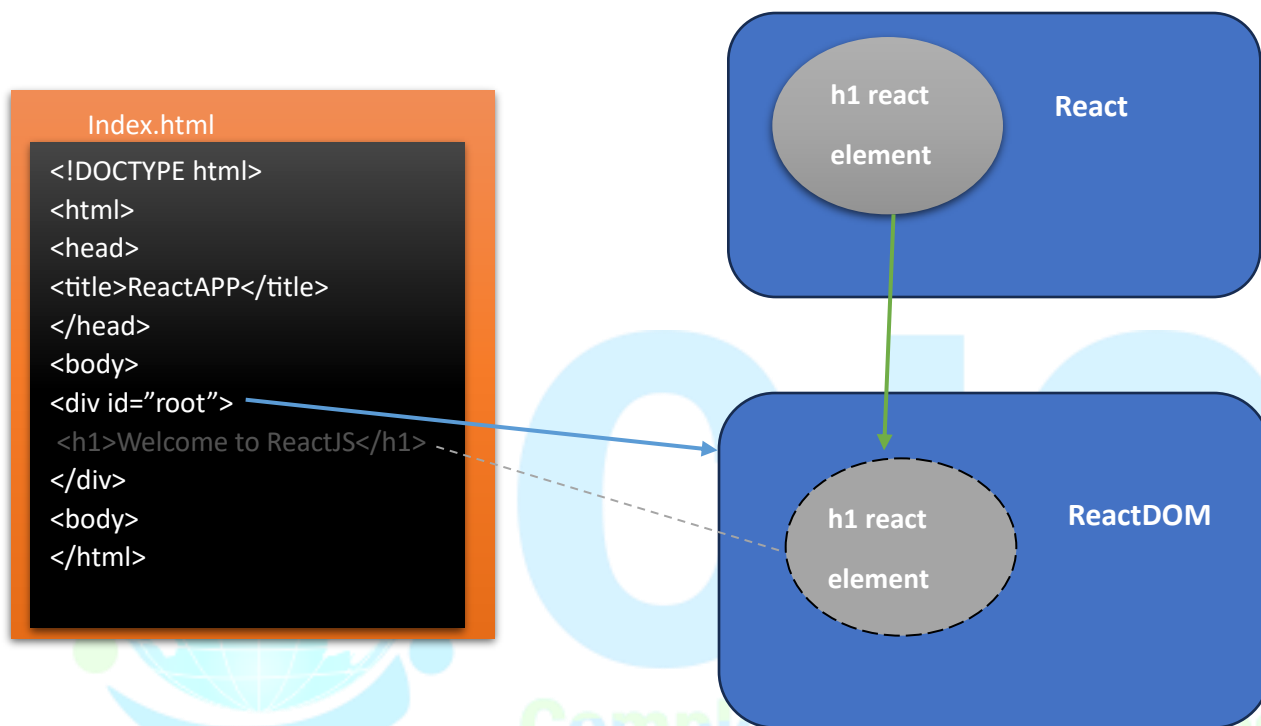   ReactDOM.render(element , document.getElementById("root"));

render( ) function have two parameter arguments.

  1st argument: - React element which need to render.

  2nd argument: - reference of dom element where we want to add our react element.

by Kunal Sir

Q) Create one simple h1 element using React and ReactDOM library.


```
Index.html
<!DOCTYPE html>
<html>
<head>
<title>ReactAPP</title>
</head>
<body>
<div id="root">
  <h1>Welcome to ReactJS</h1>
</div>
<body>
</html>
```

Note :- To use React and ReactDOM libraries add their CDN links in html page.

```
<script crossorigin
src="https://unpkg.com/react@18/umd/react.development.js"></script>

<script crossorigin src="https://unpkg.com/react-dom@18/umd/react-dom.development.js"></script>
```

```
<!DOCTYPE html>
<html lang="en">
<head>
```

```html
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <script crossorigin src="https://unpkg.com/react@18/umd/react.development.js"></script>
    <script crossorigin src="https://unpkg.com/react-dom@18/umd/react-dom.development.js">
    </script>
    <title>Document</title>
</head>
<body>
    <div id="root">

    </div>
    <script>
        // Creating h1 React Element
        let h1Element=React.createElement("h1",{id:"heading"},"Welcome To ReactJs.");

        // Adding h1 react element to the DOM Using ReactDOM library.
        ReactDOM.render(h1Element,document.getElementById("root"));

    </script>
</body>
</html>
```

Task: - Create One div element using react library and add some inline CSS to it. Render this react div element in "root" dom element using ReactDOM.

Problem statement: -

Creating User-interfaces with React is as complex as creating user-interfaces with JavaScript.

by Kunal Sir

Q) What is JSX and why was it introduced?

- JSX stands for JavaScript SML Extension.
- JSX allows us to write HTML in React.
- JSX makes it easy to write and add HTML in react.
- JSX is very similar to the HTML.
- Before JSX, Create User-interfaces with React is as complex as creating user-interfaces with JavaScript.

Q) What is babel compiler?

Babel compiler is use in react to compile the JSX code, and convert it into react elements.

```
<script src="https://unpkg.com/@babel/standalone/babel.min.js"></script>
```

```html
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <script crossorigin src="https://unpkg.com/react@18/umd/react.development.js"></script>
    <script crossorigin src="https://unpkg.com/react-dom@18/umd/react-dom.development.js"></script>
    <script src="https://unpkg.com/@babel/standalone/babel.min.js"></script>
    <title>Document</title>
</head>
<body>
    <div id="root">
    </div>
    <script type="text/babel">
        var btn=<h1>Welcome to Easy ReactJS </h1>;
        ReactDOM.render(btn,document.getElementById("root"));
    </script>
</body>
</html>
```