React

- ✓ **ReactJS** is an open-source front-end JavaScript library for building user interfaces.
- ✓ **ReactJS** is maintained by Facebook and developer's community.
- ✓ It is widely used for building single-page applications and mobile applications.
- ✓ It is very easy to use, and it allows users to create reusable UI components.

Features of ReactJS:

- ✓ **JSX** : JSX is an extension to JavaScript. Though it is not mandatory to use JSX in react, it is one of the good features and easy to use.
- ✓ **Components**: Components are like pure JavaScript functions that help make the code easy by splitting the logic into reusable independent code. We can use components as functions and components as classes. Components also have a state, props which makes life easy. Inside a class, the state of each of the props is maintained.
- ✓ **Virtual DOM**: React creates a virtual dom, i.e., in-memory data-structure cache. Only the final changes of DOM has later updated in the browsers DOM.
- ✓ **JavaScript Expressions**: JS expressions can be used in the JSX using curly brackets {}.

Advantages of ReactJS:

- ✓ ReactJS uses virtual Dom that makes use of in-memory data-structure cache, and only the final changes are updated in browsers dom. This makes the app faster.
- ✓ You can create components of your choice by using the react component feature. The components can be reused and also helpful in code maintenance.
- ✓ ReactJS is an open-source JavaScript library, so it is easy to start with.
- ✓ ReactJS has become very popular in a short span and maintained by Facebook and Instagram. It is used by many famous companies like Apple, Netflix, etc.
- ✓ Facebook maintains ReactJS, the library, so it is well maintained and kept updated.
- ✓ ReactJS can be used to develop rich UI for both desktop and mobile apps.
- ✓ Easy to debug and test as most of the coding is done in JavaScript rather than on Html.

Disadvantages of ReactJS:

- ✓ Most of the code is written in JSX, i.e., HTML and CSS are part of JavaScript, it can be quite confusing as most other frameworks prefer keeping Html separate from the JavaScript code.
- ✓ The file size of ReactJS is large.

Environment for React:

1. download & install: **VS code** (visual studio code Software)
2. download & install: **Nodejs** (Software) visit-https://nodejs.org/en/download/
3. check version of node and npm in Command prompt:
    - C:\Users\admin>**node –v**
    - C:\Users\admin>**npm –v**

React Installation Steps:

Step 1: Install React on your System by using npm command:

```
npm install –g create-react-app
```

Step 2: Create React Project by using npx command:

```
npx create-react-app project_name
```

Step 3: Open project Folder in VS Code and Run the project by using npm command:

```
npm start
```

Folder Structure:

1. **node_modules:** It contains the React library and any other third party libraries.
2. **public:** It holds the public assets of the application. It contains the index.html where React will mount the application by default on the element.
3. **src:** It contains the App.css, App.js, App.test.js, index.css, index.js, and serviceWorker.js files. Here, the App.js file always responsible for displaying the output screen in React.
4. **package-lock.json:** It is generated automatically for any operations, where npm package modifies either the node_modules tree or package.json.
5. **package.json:** It holds various metadata required for the project. It gives information to npm, which allows to identify the project as well as handle the project's dependencies.
6. **README.md:** It provides the documentation to read about React.

Rendering an Element into the DOM:

✓ Create a React root for the supplied container and return the root. The root can be used to render a React element into the DOM with render:

```
const root = ReactDOM.createRoot(container);
```

✓ React renders HTML to the web page by using a function called render().

```
root.render(element);
```

e.g:

```
const root = ReactDOM.createRoot(document.getElementById('root'));
root.render(
      <h1>Welcome to React App</h1>
 );
```