**What is React.js?**

React.js is a declarative, efficient, and flexible JavaScript library for building user interfaces.It is usually used to create view layers for website and mobile apps.

**How to set up required tools to develop web applications with React?**

* Your familiarity with HTML and JavaScript
* If you’re learning React or creating a new single-page app, use Create React App.
* If you’re building a server-rendered website with Node.js, try Next.js.
* If you’re building a static content-oriented website, try Gatsby.
* If you’re building a component library or integrating with an existing codebase, try More Flexible Toolchains.

**What is React JSX?**

React JSX is a JavaScript XML syntax, which allows React to write HTML elements and place them in the DOM without any createElement() and/or appendChild() methods.

**What are components in react?**

Components in react serve the same purpose as JavaScript functions, but work in isolation and return HTML. We use components to tell React what we want to see on the screen.

**How to create a new react app?**

* Install the latest version of Node.js
* Open Node.js and type these command to create a new app
  + npx create-react-app my-app
  + cd my-app
  + cd src
  + # If you're using a Mac or Linux:
  + rm -f \*
  + # Or, if you're on Windows:
  + del \*
  + # Then, switch back to the project folder
  + cd ..
  + import React from 'react';
  + import ReactDOM from 'react-dom/client';
  + import './index.css';
* Run npm start in the project folder and open <http://localhost:3000>

**What is the react state?**

React state is a component that maintains internal state data, allowing an object in a component to be able to be used, encapsulated and changed.

**Demonstration of state in react app**

class Car extends React.Component {

constructor(props) {

super(props);

this.state = {brand: "Ford"};

}

render() {

return (

<div>

<h1>My Car</h1>

</div>

);}

}

**What is an event?**

Just like HTML DOM events, React events can be used to perform actions based on user events on the view page; such as: click, change, mouseover etc.

**How are events handled in react?**

Handling events with React elements is very similar to handling events on DOM elements. There are some syntax differences:

* React events are named using camelCase, rather than lowercase.
* With JSX you pass a function as the event handler, rather than a string.

**Demonstration of react event**

class LoggingButton extends React.Component {

handleClick = () => {

console.log('this is:', this);}

render() {

return (

<button onClick={this.handleClick}>Click me</button>

);}

}

**What are react hooks?**

React hooks allow function components to have access to state and other React features.Therefore, class components are generally no longer needed.

**Demonstration of using hooks**

**function ExampleWithStateHook() {**

**const [Hook, setHook] = useState('hook');**

**}**

**How to build a single page web application with react.js?**

We can build a single page web application by:

* Build the displaying Initial Frame
* Create each of our individual “pages” with separate components that will feed into the main component.
* Create routing and navigating functions to connect the components to each other.