# React Js

# Module : - 9

**React Intro**

**1** :- **What is React Js?**

**Ans : - ReactJS is a JavaScript library** used for building **reusable UI components**.

React is a JavaScript library for building user interfaces.

React is used to build **single-page applications.**

**2:- What is NPM in React js ?**

**Ans :-** NPM stands for Node Package Manager. It’s tool used in ReactJs to manage and install packages. With NPM , You can easily add, update, and remove packages from your projects. It’s a convenient way to handle dependencies and streamline your development process.

**3:- What is Role of Node Js in react Js?**

**Ans :-** Node.js plays a crucial role in React.js development. It provides a runtime environment for executing JavaScript code outside of a web browser. With Node.js, React developers can use tools like NPM to manage dependencies, build and bundle their React applications, and run server-side code. Node.js enables server-side rendering, API integrations, and server-side logic in React.js applications. It's an essential component for full-stack development with React.js**.**

**4:- What is CLI command In React Js?**

**Ans :-** In React.js, CLI stands for Command Line Interface. It's a tool that allows developers to interact with and manage their React projects through the command line. The React CLI provides various commands for creating new projects, starting development servers, building production-ready code, and more. It simplifies the development process by automating common tasks and providing a streamlined workflow for React.js projects.

**5 :- What is Components in React Js?**

**Ans : -**  In React.js, components are the building blocks of a user interface. They are reusable and self-contained pieces of code that encapsulate a specific functionality or UI element. Components can be either functional or class-based. Functional components are simple functions that return JSX, while class-based components are JavaScript classes that extend the React.Component class. Components allow developers to create modular and reusable code, making it easier to manage and maintain complex UI structures in React.js applications.

**6 :- What is Header and Content Components in React Js?**

**Ans :-** In React.js, the Header and Content components are commonly used to structure the layout of a web page or application.

The Header component typically contains elements such as the site logo, navigation menu, and other header-related content. It provides a consistent and easily identifiable section at the top of the page.

The Content component, on the other hand, holds the main content of the page. It can include various components, such as text, images, forms, or other UI elements, depending on the specific content and functionality of the page.

By separating the header and content into distinct components, it becomes easier to manage and update different sections of the page independently, promoting code reusability and maintainability.

**7 :- How to install React Js on Windows, Linux Operating System? How to Install NPM and How to check version of NPM?**

**Ans :-**

* To Install react.js requires Node.js to run. You can download and install Node.js from the official website.
* Open a terminal or command prompt
* Install create-react-app. This is a command-line tool that sets up a new React.js project with the necessary files and dependencies. Run thr following Command: npm install-g-create-react-app
* Create a new project React.js Navigate to the directory where you want to create your project and run the following command: npx create-react-app my-app
* Once the project is created, Navigate into the project directory: cd my-app
* Start the development server: Run th following command to start the development server and open your react.js application in the browser: npm start

To install npm, you don’t need to do anything separately as it comes bundled with Node.js

To check the version of npm, you can run the following command in your terminal or command prompt: npm –v

This will display the installed version of npm.

Example: $npm –v

7.24.0

**8 :- How to check version of React Js?**

**Ans :-** To check the version of React.js in your project, you can open the `package.json` file located in the root directory of your React.js project. Inside the `package.json` file, you will find a `"react"` dependency entry with its corresponding version number.

Here's an example of what the `"dependencies"` section in a `package.json` file might look like:

json

"dependencies": {

"react": "17.0.2",

"react-dom": "17.0.2"

}

**9 : - How to change in components of React Js?**

**Ans :-** To make changes to components in React.js, you can follow these steps:

**1**. Locate the component you want to change: Components in React.js are typically stored in separate files with a `.js` extension. Find the file that contains the component you want to modify.

**2.** Open the component file: Use a code editor to open the component file. You can make changes to the component's code within this file.

**3.** Make your desired changes: Modify the component's code according to the changes you want to make. You can update the component's structure, add or remove elements, update styling, or modify the component's behavior.

**4.** Save the changes: After making the necessary modifications, save the component file.

**5.** Check the changes in your application: If you have your React.js application running in development mode (`npm start`), the changes should automatically be reflected in your application. If not, stop and restart the development server (`npm start`) to see the updated component.

By following these steps, you can make changes to components in React.js and see the updated results in your application.

**10 : - How to Create a List View in React Js?**

**Ans :-**

- Create a list of elements in React in the form of an array and store it in a variable.

- We will render this list as an unordered list element in the browser.

- We will then traverse the list using the JavaScript map() function and updates elements to be enclosed between <li> </li> elements.

- Finally we will wrap this new list within <ul> </ul> elements and render it to the DOM.

**Example :-**

import React from 'react';

import ReactDOM from 'react-dom';

const numbers = [1,2,3,4,5];

const updatedNums = numbers.map((number)=>{

return <li>{number}</li>;

}); ReactDOM.render( <ul>

{updatedNums}

</ul>,

document.getElementById('root')

);

**11 :- Create Increment decrement state change by button click?**