MODULE: 2 (JavaScript Essentials)

**MODULE: 2 (JavaScript Essentials)**

[Year]

DELL

[Type the company name]

[Pick the date]

**2) What is JavaScript Output method?**

**Ans.**

1. **document.write()**: This method is used to write HTML expressions or JavaScript code to a document. It is mostly used for testing purposes or for inserting dynamic content into a web page.
2. **console.log()**: This method is used to output data to the console of the browser's developer tools. It is commonly used for debugging and troubleshooting purposes.
3. **window.alert()**: This method is used to display a message in a dialog box. It is commonly used to display error messages or to prompt the user for input.
4. **innerHTML**: This property is used to write HTML content inside an element. It is commonly used to dynamically update the content of a web page.
5. **document.writeLn()**: This method is used to write text followed by a newline character to a document. It is mostly used for outputting formatted text.

**3). How to used JavaScript Output method?**

**Ans.**

<https://github.com/chandaniacharya/React-JS/tree/main/javascript%20assencial%20assignment%20ques/javscript%20output%20method%20q3>

**4). How to used JavaScript Events to do all examples?**

**Ans.**

<https://github.com/chandaniacharya/React-JS/tree/main/javascript%20assencial%20assignment%20ques/js%20eventsq4>

**MODULE: 3 (JavaScript Essentials)**

1. **What is React Js?**

**Ans.**

React is a JavaScript library for building user interfaces. It was developed by Facebook and has gained a lot of popularity in recent years due to its performance, flexibility, and reusability.

React allows developers to create reusable UI components that can be easily combined to create complex and interactive interfaces. It uses a virtual DOM (Document Object Model) to efficiently update the UI without the need for full page reloads, which improves performance and reduces the complexity of managing state changes.

**2). What is NPM in React Js?**

**Ans.**

React applications are typically built using a tool called Create React App, which is built on top of Node.js and uses NPM to manage dependencies. When a developer starts a new React project using Create React App, NPM is used to install all the necessary packages and libraries required to get the project up and running.

NPM also provides a command-line interface that developers can use to perform various tasks related to managing packages and dependencies, such as installing new packages, updating existing packages, and removing packages that are no longer needed.

**3). What is Role of Node Js in react Js?**

**Ans**.

Node.js is a server-side JavaScript runtime environment, which means it allows JavaScript code to run on the server-side rather than just in the browser.

In the context of React.js, Node.js is often used as a development tool for building and testing React applications. Node.js comes with a package manager called npm (Node Package Manager) that is used to manage dependencies and packages for React applications. Developers can use npm to install React and other third-party packages required for building a React application.

**4.) What is CLI command In React Js?**

**Ans**

. In React.js, CLI (Command Line Interface) commands are used to perform various tasks related to building, testing, and deploying React applications.

Some common CLI commands used in React.js development include:

* **npx create-react-app my-app**: This command creates a new React application called "my-app" using the Create React App tool.
* **npm start**: This command starts the development server and opens the application in the default browser. It also automatically reloads the application when changes are made.
* **npm run build**: This command creates a production-ready build of the React application by compiling and optimizing the code.
* **npm test**: This command runs the test suite for the React application.
* **npm run eject**: This command "ejects" the React application from the Create React App environment and exposes all configuration files and build scripts.

**5). What is Components in React Js?**

**Ans.**

Components are the building blocks of React.js applications. In React.js, a component is a self-contained piece of code that defines the UI of a portion of the application. Components can be thought of as reusable templates for creating UI elements such as buttons, forms, or navigation menus.

components are classes that extend the **React.Component** class and define a **render()** method that returns a UI element.

Overall, components are a fundamental concept in React.js and are essential for building modular, reusable, and maintainable user interfaces.

**6). What is Header and Content Components in React Js?**

**Ans**.

Header and Content components are common types of components used in React.js applications to define the layout and structure of the user interface.

Overall, the Header and Content components are simple examples of how components can be used to build the layout and structure of a React.js application.

<https://github.com/chandaniacharya/React-JS/tree/main/MODULE3%20JS%20assencial>

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

**Ans.**

Download and install Node.js from the official website: <https://nodejs.org/en/download/>

1. Open a terminal or command prompt.
2. Run the following command to install ReactJS globally:

npm install -g create-react-app

1. Once the installation is complete, you can create a new React project by running the following command:

create-react-app my-app

Replace **my-app** with the name of your project.

1. Navigate to your project directory and start the development server:
2. cd my-app

npm start

8). How to check version of React Js

Ans. To check the version of ReactJS installed in your project, you can open a terminal or command prompt, navigate to your project directory, and run the following command:

npm view react version

This will display the version number of ReactJS installed in your project.

If you want to check the global version of ReactJS installed on your system, you can run the following command:

npm list -g react

This will display the version number of ReactJS installed globally on your system.

**9). How to change in components of React Js?**

**Ans.**

1. Locate the component you want to make changes to. Components are usually located in the **src/components** directory in a typical ReactJS project.
2. Open the file containing the component code in your preferred code editor.
3. Make the necessary changes to the component code. This can include changing the component's JSX code, modifying the component's state or props, or adding new methods to the component.
4. Save the changes to the file.
5. If the changes you made affect the appearance of the component, refresh the browser window where the application is running to see the changes.

**10). Create Increment decrement state change by button click**

**Ans.**

import React, { useState } from "react";

function Counter() {

  const [number, setNumber] = useState(0);

  const handleIncrement = () => {

    setNumber(number + 1);

  };

  const handleDecrement = () => {  // Define function to handle increment button click

    setNumber(number - 1);

  };

  return (

    <div>

      <button onClick={handleIncrement}>+</button>

      <h1>{number}</h1>

      <button onClick={handleDecrement}>-</button>

    </div>

  );

}

export default Counter;