Q1) What is React Js?

Answer:-

React Js is an open-source library for building user interfaces. It is a front-end technology that is used to develop web applications. Developed by Facebook, React helps developers to create reusable UI components and manage the state of the application in a more predictable way.

React uses a declarative syntax, which makes it easier to understand and maintain code. It is based on the concept of components, which are self-contained pieces of code that can be reused across different parts of the application.

Some of the key features of React Js include:

- Virtual DOM for improved performance
- Component-based architecture for modularity
- JSX, a syntax extension for JavaScript
- Unidirectional data flow for easier debugging
- Hooks for managing state in functional components

With its robust features and growing community support, React Js has become one of the most popular front-end technologies in the world of web development.

Q2) What is NPM in React Js?

Answer:-

NPM in React JS stands for Node Package Manager and it is a tool used to manage packages, or modules, of JavaScript code that are commonly used in web development. These packages can include libraries and frameworks that make the development process faster and more efficient. NPM allows developers to easily install, update, uninstall and publish these packages with just a few commands in the terminal.

Q3) What is Role of Node Js in react Js?

Answer:-

Node JS is a server-side platform built on Google Chrome's V8 JavaScript engine. In React JS, Node JS serves as the runtime environment for building and serving applications. It allows developers to use JavaScript on both client and server sides of web development. With this technology, React applications can be executed in the browser or on the server side which makes it faster and more efficient than traditional web development frameworks. Additionally, Node JS provides access to thousands of open-source packages that aid in developing complex functionality within the application with ease.

Q4) What is CLI command In React Js?

Answer:-

CLI stands for Command Line Interface in React JS. It is a tool that enables developers to perform various operations such as creating, building and managing applications with the help of commands typed into the terminal. CLI allows developers to automate repetitive tasks and helps them focus on coding rather than configuring. It also provides built-in features like Live Reload which reduces development time significantly by quickly updating changes made by the developer without having to refresh the page manually every time.

Q5) What is Components in React Js?

Answer:-

In React JS, components are the building blocks of any application. Simply put, a component is a piece of code that encapsulates functionality and presentation logic in one place. Components can be reused throughout an application and can be added to other applications as well. They enhance modularity by breaking up complex functionality into smaller pieces that are easier to manage and maintain over time. React allows developers to create their own custom components or use pre-built ones, making development faster and more efficient.

Q6) What is Header and Content Components in React Js?

Answer:-

In React JS, Header and Content components refer to specific parts of an application. The Header component typically contains information such as branding or navigation links while the Content component displays the main content of the page. Both components can be customized to fit a particular design or functionality requirements. By breaking up these sections into their own components, developers can focus on creating reusable code rather than recreating similar elements throughout an application. This promotes consistency in design and streamlines development processes over time.

Q7) How to install React Js on Windows, Linux Operating System? How to Install NPM and How to check version of NPM?

Answer:-

To install React JS on Windows or Linux, developers can use the Node Package Manager (NPM) which is distributed with Node.js. First, you must download and install Node.js from their website. Once installed, open a terminal window and run the command 'npm i -g create-react-app'. This installs a tool that generates new React projects with ease.

To install NPM itself, follow these steps:
1. Download and install Node.js
2. Open a terminal window
3. Type "npm -v" to check if NPM is already installed
4. If not installed, type "sudo apt-get update"
5. Then type "sudo apt-get npm"

To check the version of NPM that's currently installed on your machine:
1. Open a terminal window
2: Type "npm -v"
3: The current version will be displayed in the output.

Q8) How to check version of React Js?

Answer:-

To check the version of React JS that's currently installed on your machine:
1. Open a terminal window
2: Navigate to the directory where your React project is located
3. Type "npm list react" or "npm view react version"
4: The current version will be displayed in the output.

Q9) How to change in components of React Js?

Answer:-

To make changes to components in React JS, developers will need to modify the code within those specific components. This can include changing the styles and layout of elements or updating the functionality of certain features. To achieve this, they can use a variety of tools such as CSS styling libraries or state management techniques like Redux or MobX. Additionally, debugging tools like React Developer Tools for Chrome can be helpful for making sure all components are functioning properly before deployment. Overall, modifying React JS components requires a strong understanding of both front-end development practices and the core principles behind React concepts such as state and props.

Q10) How to Create a List View in React Js?

Answer:-

To create a list view in React JS, developers can use the map() method to iterate over an array of items and dynamically generate individual components for each item. This allows for easy creation and management of lists without needing to hard-code every item.

Here is an example code snippet:

```
const myList = [“Item 1”, “Item 2”, “Item 3”];

function ListView () {
  return (
    <ul>
      {myList.map((item) =>
        <li key={item}>{item}</li>
      )}
    </ul>
  );
}
```

In this example, we define our list as an array called `myList`. We then define the `ListView` component which returns `<ul>` element containing a dynamic list generated by mapping over each item in `myList`. The `key` prop is included on each rendered `<li>` element to give React a stable identity when it updates the DOM later.

Q11) Create Increment decrement state change by button click?

Answer:-

To create an increment and decrement state change by button click in React JS, you can define a function that updates the state variable with new values whenever the corresponding buttons are clicked. It would look something like this:

```
import React, { useState } from 'react';

function StateChange () {
  const [count, setCount] = useState(0);

  return (
    <div>
      <button onClick={() => setCount(count + 1)}>Increment</button>
      <button onClick={() => setCount(count - 1)}>Decrement</button>
      <p>Current Count: {count}</p>
    </div>
  );
}
```

In this example code snippet, we use the `useState` hook to declare a new state variable called `count` starting at zero. We then render two buttons for incrementing and decrementing the count value respectively. Each button's `onClick` handler is passed in as an arrow function that calls `setCount` with either an incremented or decremented value of the current count. Finally, we render out the current count value within a paragraph element using interpolation syntax (`{}`) to display it dynamically on screen.