1) What is React Js?

Ans> React Js, in simple terms, is a JavaScript library that helps developers build user interfaces for web applications. It's like a set of building blocks for making websites. It helps developers build interactive and dynamic web pages by breaking them into reusable components.

2) What is NPM in React Js?

Ans> When working with React Js, developers often use npm to install and manage packages, such as React itself, as well as other libraries and tools needed for their projects. Npm simplifies the process of adding functionality to React applications by providing a centralized repository of packages that can be easily installed and updated.

3) What is Role of Node Js in React Js?

Ans> In simple terms, Node.js supports React.js development by providing tools and functionalities such as:

1. Server-side rendering: Node.js helps React applications render on the server-side, which improves performance and SEO.
2. Development server: It runs a local server for testing React apps, enabling real-time updates during development.
3. Build tools: Node.js facilitates bundling and transpiling React code using tools like webpack and Babel.
4. Package management: It includes npm for managing dependencies and libraries used in React projects.

4) What is CLI command in React Js?

Ans> In simple terms, a CLI (Command Line Interface) command in React.js is a way to interact with and control your React.js project using text-based commands entered into a terminal or command prompt.

For example, when you create a new React.js project using a tool like Create React App, you typically use a CLI command like "npx create-react-app my-app". This command sets up a new React.js project folder structure and installs necessary dependencies automatically.

5) What is Components in React Js?

Ans> In simple terms, components in React.js are like building blocks for creating web interfaces. They are reusable pieces of code that represent different parts of a user interface, such as buttons, forms, or headers.

For example, you might have a "Navbar" component for the navigation bar at the top of your website, a "Button" component for buttons throughout your application, and a "Form" component for input fields and submission buttons.

By breaking your UI into components, you can build large and complex applications more efficiently, as you can reuse and compose these components to create different layouts and functionalities.

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

Ans> In simple terms, Header and Content components in React.js are two types of building blocks used to create web interfaces.

1. Header Component: This component represents the top section of a webpage, typically containing a logo, navigation links, and maybe some other information like user login/logout options. It's like the header of a document, providing a way for users to navigate around the site and access important features.
2. Content component: This component represents the main content area of a webpage, where most of the page's content is displayed. It could include things like articles, images, videos, or any other type of information relevant to the page's purpose. Essentially, it's the heart of the webpage, displaying the core information or functionality that users are interested in.

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

Ans> Installing React.js and npm on Windows and Linux is quite similar. Here are the steps:

1. Install Node.js: React.js is typically installed and managed using npm, which comes bundled with Node.js. You can download and install Node.js from the official website: Node.js Downloads.
2. Verify Installation: After installing Node.js, you can verify that both Node.js and npm are installed correctly by opening a terminal or command prompt and running the following commands: node-v and npm-v.
3. Install Create React App: To install Create React App, run the following command: npm install -g create-react-app
4. Create a New React App: Once Create React App is installed, you can create a new React.js project by running: npx create-react-app my-app

Cheking npm version: To check the version of npm installed on your system, simply run the following command in your terminal or command prompt: npm -v

8) How to check version of React Js?

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

npm list react

9) How to change in components of React Js?

Ans> In simple words, to change components in a React.js application, follow these steps:

1. Find the Component: Locate the component file you want to change in your project's folder structure.
2. Edit the code: Open the component file with a text editor. Make the desired changes to the code. You can adjust how it looks or behaves.
3. Save the Changes: After editing, save the file.
4. See Changes: If you're running your React app, the changes will appear in the browser. If not, start or refresh your app to see the updates.
5. Test: Make sure the component still works as intended after the changes. If not, go back and adjust until it does.
6. Repeat as Needed: If you need to make more changes, go back to step 2.