

Interview Questions on Full Stack Development

Infosys SpringBoard

What is CDN?

CDN stands for Content Delivery Network. It is a distributed network of servers that provides faster delivery of website content by caching it on multiple servers.

What is a Cross Origin Attribute??

A Cross Origin Attribute is a security feature used in web development that allows or restricts scripts, stylesheets, images, and other resources from different domains to interact with each other.

What is Babel?

Babel is a JavaScript transpiler that converts modern JavaScript code into a backward-compatible version that can run in older browsers. It is commonly used in web development to ensure compatibility across different platforms.

What is webpack?

Webpack is a module bundler for JavaScript applications. It takes modules with dependencies and generates a single bundle that can be loaded by a browser.

Can you name some other bundler?

Some other popular module bundlers are Browserify, Rollup, Parcel, and FuseBox.

What is Package.json and why is it important?

package.json is a configuration file in Node.js projects that contains metadata about the project, such as dependencies, scripts, and other project-specific details. It is important because it helps in managing the project's dependencies and automating common tasks.

What is NPM?

NPM stands for Node Package Manager. It is a package manager for Node.js projects that allows developers to install and manage third-party packages and dependencies.

What is GIT?

Git is a version control system used in software development to manage code changes and collaborate with other developers. It allows developers to track and revert changes, work on different branches, and merge code changes.

What is a polyfill?

A polyfill is a piece of code that provides functionality that is not natively supported by a browser. It is commonly used to add support for new web APIs and features to older browsers.

What is React?

React is a JavaScript library for building user interfaces. It allows developers to create reusable UI components and manage the state of the application efficiently.

Why do we call React a Javascript Library?

We call React a JavaScript library because it is a collection of pre-written JavaScript code that developers can use to build user interfaces for web applications. React provides a set of reusable UI components that can be used to build complex interfaces quickly and efficiently, without having to write all the code from scratch.

Why is React so fast?

React is fast because it uses a virtual DOM (Document Object Model) to update the UI. The virtual DOM is a lightweight copy of the real DOM, and React uses it to track changes to the UI without updating the actual DOM until it's necessary. This approach makes updates faster and more efficient since React only updates the parts of the UI that have changed, rather than the entire UI. Additionally, React uses a process called reconciliation to determine the minimum number of changes required to update the UI, further improving performance. Finally, React provides tools such as code splitting and lazy loading, which help to reduce the initial load time of the application and improve performance.

What is TypeScript?

TypeScript is a superset of JavaScript that adds optional static typing and other features to the language. It is commonly used in large-scale JavaScript projects to improve code quality and maintainability.

What is Reconciliation in react ?

Reconciliation is the process in which React updates the DOM based on changes in a component's state or props. It compares the old and new virtual DOM trees, applies changes only where necessary, and uses algorithms like key-based reconciliation and memoization to optimize the process and improve performance.

Can you explain the concept of a Virtual DOM in React, and how it contributes to performance?

The Virtual DOM is a lightweight copy of the actual DOM that React uses to keep track of changes in the UI. When a component's state or props change, React creates a new Virtual DOM tree, compares it to the previous tree, and identifies the specific changes that need to be made to the actual DOM. This allows React to update only the necessary parts of the UI, rather than the entire tree, which contributes to better performance. The Virtual DOM also allows for efficient batch updates and reduces the frequency of expensive DOM operations.

Can you explain the concept of React SSR (Server-Side Rendering), and how it can be used to improve SEO and page load times?

React Server-Side Rendering (SSR) involves rendering React components on the server-side and sending the fully-rendered HTML to the client. This approach improves SEO and initial page load times by allowing search engines to crawl and index the fully-rendered HTML content, and by reducing the time needed to download and parse JavaScript files on the client-side.

Can you explain the difference between server-side rendering and client-side rendering in React?

Server-side rendering (SSR) and client-side rendering (CSR) are two ways of rendering React components. SSR involves rendering React components on the server-side and sending the fully-rendered HTML to the client, while CSR involves rendering React components on the client-side browser using JavaScript. SSR is achieved by using Node.js to execute the JavaScript code and generate the HTML markup, which is sent to the client as a complete HTML page. This approach improves SEO and initial page load time, but can be slower for subsequent updates. CSR involves sending a minimal HTML page and a JavaScript bundle containing the React components and application logic to the client. The browser then downloads the bundle, executes it, and renders the components in the browser. This approach allows for greater interactivity and faster updates, but can be slower for the initial page load.

Can you explain what the DOM (Document Object Model) is, and how it is used in web development?

The Document Object Model (DOM) is a programming interface for web documents. It represents the page so that programs can change the document structure, style, and content. In other words, it creates a tree-like structure of the HTML document, allowing developers to manipulate the structure and content of the page through JavaScript. The DOM is used in web development to dynamically update the content of a web page without the need for a full page reload. JavaScript code can manipulate the DOM tree by adding, deleting, or modifying elements and attributes. This enables developers to create dynamic and interactive web pages that respond to user actions and input.

Can you state some advantage of using React over Angular in front-end applications?

the advantages of using React over Angular in point-wise answers: Performance: React uses a virtual DOM and a diffing algorithm to minimize the number of changes that need to be made to the actual DOM, resulting in faster and more efficient updates. Flexibility: React is flexible and can be used with other libraries and frameworks, allowing developers to customize their development environment and choose the tools that best suit their needs. Ease of Learning: React has a relatively small and simple API, making it easier to learn and get started with compared to Angular. Community and Ecosystem: React has a large and active community and ecosystem, with a wide range of resources and tools available, including third-party libraries and plugins. SEO-Friendliness: React makes it easier to optimize web pages for search engines as it allows for server-side rendering, which can improve website performance and SEO. Testability: React has a built-in testing library and can be easily integrated with other testing tools, making it easier to write and run tests for components. It's important to note that both React and Angular have their own unique features and strengths, and the choice between the two ultimately depends on the specific needs and requirements of the project.

React vs Angular

React

React is a small view library

React covers only the rendering and event handling part

Presentation code in JavaScript powered by JSX

React's core size is smaller than Angular, so bit fast

React is very flexible

Great performer, since it uses Virtual DOM

Angular

Angular is a full framework

Angular provides the complete solution for front-end development

Presentation code in HTML embedded with JavaScript expressions

Angular being a framework contains a lot of code, resulting in longer load time

Angular has less flexibility

Angular uses actual DOM which affects its performance

How do you create reusable and composable components in React, and what are some best practices for component design and architecture?

In React, reusable and composable components can be created by breaking down the UI into smaller, reusable building blocks. Each component should have a single responsibility and be easy to understand and maintain. Here are some best practices for component design and architecture in React: Use descriptive and meaningful names for components. Break down the UI into smaller, reusable components. Use props to pass data and callbacks between components. Keep the state and logic in the top-level parent components. Use `PureComponent` or memoization to optimize performance. Use `propTypes` or TypeScript to define the types of props and state. Use CSS modules or styled components to encapsulate styles. Use composition over inheritance to share functionality between components. Avoid using the context API or global state for everything. Write unit tests for each component to ensure they work as expected. By following these best practices, you can create reusable and composable components that are easy to understand, maintain, and test.

How do you render a React component to the DOM, and what is the root node?

```
const root = ReactDOM.createRoot(document.getElementById('root'));  
root.render( );
```

To render a React component, you create an instance of the component and use the `ReactDOM.render()` method to mount it to an existing element in the HTML file, which is typically referred to as the root node. The `render()` method replaces the contents of the root node with the React component.

How does the DOM tree structure work, and how is it related to HTML, CSS, and JavaScript?

The DOM is a tree-like structure that represents the content of an HTML document, with each node representing an element, attribute, or text. HTML defines the basic structure, CSS styles the content, and JavaScript can manipulate the DOM to modify the content, style, or behavior of the page. The DOM tree is created when the HTML document is parsed by the browser.

How is a node of a DOM tree updated in an application built in React?

When the state or props of a component change in React, a new virtual DOM tree is created, which is compared to the previous tree using a diffing algorithm. React identifies the minimal set of changes required to update the actual DOM and uses reconciliation to update only the corresponding nodes that have changed. This process minimizes the number of actual DOM operations that need to be performed.

What is a diffing algorithm in React?

A diffing algorithm in React is a process that compares the current virtual DOM tree with the previous one and identifies the minimal set of changes required to update the actual DOM. This is done by checking the type, props, and children of each node in the tree and updating only those specific parts of the DOM that need to be changed, rather than updating the entire tree.

What is a SPA or Single Page Application?

A Single Page Application (SPA) is a web application that dynamically updates a single HTML page using JavaScript frameworks, such as React, Angular, or Vue.js. The initial HTML file, JavaScript, and CSS are loaded when the user visits the application, and subsequent interactions with the application trigger updates to the page content without requiring a full page reload. This provides a seamless and responsive user experience.

What is React, and how does it differ from other front-end frameworks or libraries?

React is a JavaScript library for building user interfaces. It is different from other front-end frameworks or libraries because it focuses only on the view layer, uses a declarative programming style, and has a large and active community.

What is Emmet?

Emmet is a web development tool that helps in writing HTML and CSS code quickly and efficiently. It uses abbreviations or shortcuts to generate code.