1. **What is dependency and dev dependency? What's the difference?**
   A: A software dependency refers to a bundle of code (such as a library) that provides our code functionality so that it can run. Our code depends on this code to run, which is why it's called a dependency. Dev dependencies are only necessary for the development phase, and not necessary for running the application.

2. **What is package.json and package-lock.json? What's the difference?**
   A: package-lock.json locks in all our dependencies to their specific version number, allowing dependency trees to be built with future installs. package.json can select version ranges, and records the bare minimum of the direct dependencies, and also stores other necessary metadata.

3. **What is JSX? Why do we need babel?**
   A: JSX is the syntactic sugar that allows us to write in a format that is more familiar to developers that are used to coding with vanilla HTML, CSS, and JS by embedding HTML-like syntax directly into our JavaScript code. Browsers do not understand JSX, however, so Babel helps translate JSX into JavaScript.

4. **SPA vs MPA**
   A: Single-page applications have every different "page" in a web application load on the same page. In other words, the same page is dynamically updating content through JavaScript, which leads to a faster and smoother transition. Multiple-page applications are more traditional, wherein each page is a separate HTML file. They are great for search engine optimization, and have a faster initial load time.

5. **CSR vs SSR**
   A: Client-side rendering is a rendering technique in which after a user makes a request, the server will send a HTML file with minimal content that has links to CSS and JavaScript, and the page is rendered by the client fetching data from the JavaScript file. This is advantageous for web interactivity and reducing the server load, but it can limit SEO and have a lengthier initial loading time.

   Server-side rendering occurs on the server, generating the complete HTML file with rendered UI before sending it to the client's browser. This is the traditional method, and so it has the opposite advantages as CSR: faster initial loading time and great for SEO.