Common Questions and Answers for MERN Stack

React

Q: What is React?

A: React is a JavaScript library for building user interfaces, maintained by Facebook.

Q: What are components in React?

A: Components are the building blocks of React applications, representing parts of the UI.

Q: What is the difference between class and functional components?

A: Class components use ES6 classes, while functional components are simpler and support hooks.

Q: What is JSX?

A: JSX is a syntax extension for JavaScript, allowing HTML-like code in React.

Q: What are props in React?

A: Props are inputs to components that allow data to be passed between them.

Q: What is state in React?

A: State is an object that represents a component's local data or UI changes.

Q: What are hooks in React?

A: Hooks let you use state and other React features in functional components.

Q: What is useEffect?

A: useEffect is a hook for performing side effects in functional components.

Q: What is React Router?

A: React Router is a library for managing navigation and routing in React apps.

Q: How do you optimize React performance?

A: Use techniques like memoization, lazy loading, and avoiding unnecessary renders.

Node.js

Q: What is Node.js?

A: Node.js is a runtime environment for executing JavaScript outside the browser.

Q: What is the purpose of npm?

A: npm is the package manager for Node.js, used to install and manage packages.

Q: What is a callback function in Node.js?

A: A callback is a function passed as an argument to another function.

Q: What is the difference between synchronous and asynchronous operations?

A: Synchronous operations block the thread, while asynchronous operations use callbacks or promises.

Q: What are streams in Node.js?

A: Streams are objects for reading or writing data in chunks.

Q: What is middleware in Express?

A: Middleware functions are executed during request processing in Express.

Q: What are the global objects in Node.js?

A: Examples include __dirname, __filename, process, and console.

Q: What is the Event Loop in Node.js?

A: The Event Loop processes asynchronous operations in Node.js.

Q: What are modules in Node.js?

A: Modules are reusable pieces of code that can be imported and exported.

Q: What is the purpose of the package.json file?

A: package.json contains metadata about the project and its dependencies.

Express.js

Q: What is Express.js?

A: Express.js is a minimal and flexible Node.js web application framework.

Q: What is a route in Express.js?

A: Routes define the endpoints and how they respond to client requests.

Q: What is app.use() in Express.js?

A: app.use() is for mounting middleware functions globally.

Q: What is the difference between app.get() and app.post()?

A: app.get() handles GET requests, while app.post() handles POST requests.

Q: What are query parameters in Express.js?

A: Query parameters are key-value pairs in the URL after the '?'.

Q: What is the purpose of the response object?

A: The response object sends data back to the client.

Q: What is Express Router?

A: Express Router is used for modularizing routes in an application.

Q: How do you handle errors in Express?

A: Use middleware functions for centralized error handling.

Q: What is a template engine?

A: A template engine generates HTML using static files and dynamic data.

Q: What is CORS in Express?

A: CORS allows or restricts resources from different origins to be accessed.

MongoDB

Q: What is MongoDB?

A: MongoDB is a NoSQL database that stores data in BSON format.

Q: What are collections in MongoDB?

A: Collections are groups of MongoDB documents, equivalent to tables in SQL.

Q: How do you create a database in MongoDB?

A: Use the `use` command with the desired database name.

Q: What are CRUD operations?

A: CRUD stands for Create, Read, Update, and Delete operations.

Q: What is the purpose of indexes in MongoDB?

A: Indexes improve the performance of queries.

Q: What is a replica set?

A: A replica set is a group of MongoDB servers that provide redundancy and high availability.

Q: What are aggregations in MongoDB?

A: Aggregations process data records and return computed results.

Q: What is the difference between find() and findOne()?

A: find() returns multiple documents, while findOne() returns a single document.

Q: What are schemas in MongoDB?

A: Schemas define the structure of documents in a collection.

Q: How do you connect to MongoDB using Node.js?

A: Use the MongoDB driver or Mongoose library.