1. **What is Node.js?** Explain what Node.js is and its main features, such as event-driven, non-blocking I/O, and its server-side capabilities.
2. **How do you install Node.js on your system?** Describe the steps to install Node.js on your computer, and mention the role of npm (Node Package Manager).
3. **What is npm, and what is its purpose in Node.js?** Define npm and its significance in Node.js for managing packages and dependencies.
4. **Explain the concept of asynchronous programming in Node.js.** Describe how Node.js handles asynchronous operations and the role of the event loop.
5. **What are callbacks in Node.js?** Define callbacks and explain how they are used in handling asynchronous operations in Node.js.
6. **What is the role of the fs module in Node.js?** Discuss the purpose of the **fs** module and how it enables file system operations in Node.js.
7. **How do you create an HTTP server in Node.js?** Explain the process of creating a basic HTTP server using Node.js and the **http** module.
8. **What is an EventEmitter in Node.js?** Describe the EventEmitter class and its role in handling events in Node.js.
9. **How do you read data from the request body in a Node.js HTTP server?** Explain how to read data from the request body using the **data** and **end** events.
10. **What are modules in Node.js, and how do you use them?** Define modules and explain how to create, import, and export modules in Node.js using the CommonJS module system.
11. **How do you handle errors in Node.js?** Discuss common error handling techniques, including try-catch blocks and handling asynchronous errors.
12. **What are the benefits of using Express.js?** Explain the advantages of using the Express.js framework in Node.js and how it simplifies web application development.
13. **What is the purpose of the package.json file in a Node.js project?** Describe the role of the package.json file and its importance for managing project dependencies and scripts.
14. **How do you install external packages using npm?** Explain how to install and manage external packages using npm.
15. **What is the difference between require and import in Node.js?** Compare the use of **require** and **import** statements in Node.js, and when to use each.
16. **What is the role of the path module in Node.js?** Describe the purpose of the **path** module and how it helps with file path operations.
17. **How can you handle file uploads in a Node.js server?** Explain how to handle file uploads using packages like **multer** in a Node.js server.
18. **What are streams in Node.js, and how do they work?** Describe the concept of streams and how they enable efficient data handling in Node.js.
19. **How can you set up a simple API using Express.js?** Describe the steps to create a basic API with routes and endpoints using Express.js.
20. **What is middleware in Express.js?** Explain the concept of middleware in Express.js and how it allows you to perform additional operations before handling the final request.