Node.js Interview Questions and Answers

1. What is Node.js?

Node.js is an open-source, cross-platform JavaScript runtime environment that allows developers to run JavaScript code outside of a browser. It is built on Chrome's V8 JavaScript engine and is used for building scalable server-side and networking applications.

2. What are the key features of Node.js?

Some key features include:

- Asynchronous and Event-Driven
- Fast Execution using V8 Engine
- Single-threaded but highly scalable
- NPM (Node Package Manager)
- Cross-platform compatibility

3. Explain the event-driven architecture of Node.js.

Node.js uses events to handle asynchronous operations. It utilizes an event loop and callbacks to manage operations without blocking the main thread, making it suitable for I/O-heavy tasks.

4. What is the difference between synchronous and asynchronous functions in Node.js?

- Synchronous functions block the execution until the task is complete.
- Asynchronous functions do not block execution; they continue running and use callbacks/promises to handle the result once ready.

5. What is NPM?

NPM (Node Package Manager) is the default package manager for Node.js. It provides a platform for sharing and managing reusable code packages/modules.

6. What is a callback function?

A callback function is a function passed as an argument to another function, to be executed later once a task is complete, enabling asynchronous behavior.

7. What are Promises in Node.js?

Promises are objects representing the eventual completion or failure of an asynchronous operation. They help manage asynchronous flow more cleanly than nested callbacks.

8. What is async/await in Node.js?

'async/await' is syntactic sugar built on Promises, making asynchronous code look and behave like synchronous code, improving readability and error handling.

9. How does the Node.js event loop work?

The event loop is the core of Node.js's asynchronous behavior. It continuously checks the call stack and task queue, executing tasks from the queue when the stack is clear.

10. What are streams in Node.js?

Streams are objects used to read or write data continuously. Node.js has four types: Readable, Writable, Duplex, and Transform.

11. What is the difference between process.nextTick() and setImmediate()?

'process.nextTick()' queues a callback to be invoked in the same phase of the event loop, before the next I/O event. 'setImmediate()' queues it to be executed on the next iteration of the event loop.

12. What is middleware in Express.js?

Middleware functions are functions that have access to the request, response, and next middleware in the app's request-response cycle. They're used for tasks like authentication, logging, and error handling.

13. What is the difference between require and import in Node.js?

'require' is the older CommonJS module system, while 'import' is part of ES Modules (ESM). 'import' allows for static analysis and is becoming the standard in modern JavaScript.

14. What is clustering in Node.js?

Clustering is a technique to create child processes (workers) that share the same server port. This helps in utilizing multi-core systems to handle concurrent requests.

15. How do you handle errors in Node.js?

Errors can be handled using try/catch blocks for synchronous code and '.catch()' or 'try/catch' with 'async/await' for asynchronous code. Also, error-handling middleware is used in Express.