1. **Explain in brief what is node js?**

=>**Node.js** is an open-source, cross-platform, back-end JavaScript runtime environment that runs on the V8 engine and executes JavaScript code outside a web browser. Node.js lets developers use JavaScript to write command-line tools and for server-side scripting—running scripts server-side to produce dynamic web page content before the page is sent to the user's web browser

2. **How is node js non-blocking?**

=>Non-Blocking nature of node.js simply means that **node.js proceeds with the execution of the program instead of waiting for long I/O operations or HTTP requests**.

3. **What is throughput?**

=>Basically, "Throughput" is the number of transactions produced over time during a test. It's also expressed as the amount of capacity that a website or application can handle

4. **How is Node js having high IO throughput?**

=>Node.js is an open-source, cross-platform, back-end JavaScript runtime environment that runs on the V8 engine and executes JavaScript code outside a web browser. Node.js lets developers use JavaScript to write command-line tools and for server-side scripting—running scripts server-side to produce dynamic web page content before the page is sent to the user's web browser.

5. What are CPU-intensive tasks?

=>So what are CPU Intensive tasks? They are **complex user actions that eat up more RAM**. A few of such processes can shut down your server entirely.

6.**How can you end up blocking your main thread in node.js?**

=>Node js is a single-threaded and highly scalable system. Instead of separate processes and threads, it uses asynchronous, event-driven I/O operations. So It can achieve high output via single-threaded event loop and **non-blocking** I/O

7. **What is the event loop?**

=>JavaScript has a runtime model based on an event loop, which is responsible for executing the code, collecting and processing events, and executing queued sub-tasks. This model is quite different from models in other languages like C and Java.

8. **What are different phases in event loop?**

=>The Event Loop contains six main phases: **timers, I/O callbacks, preparation / idle phase, I/O polling, setImmediate() callbacks execution, and close events callbacks**.

9. **What is process.tick?**
=>**Every time the event loop takes a full trip**, we call it a tick. When we pass a function to process.nextTick() , we instruct the engine to invoke this function at the end of the current operation, before the next event loop tick starts: JS copy.

10. **When can process.tick starve your event loop?**
=>A tick is **the dequeuing of an event from the "event loop queue" and the execution of said event**.

11. **What is the difference between setTimeout and setInterval?**
=>setTimeout allows us to run a function once after the interval of time. setInterval allows us to run a function repeatedly, starting after the interval of time, then repeating continuously at that interval

12. **How can you make a network request with http module from the backend?**
=>The most common HTTP request methods have a call shortcut (such as http. get and http. post), but you can make any type of HTTP request by **setting the call field to http. request and specifying the type of request using the method field**.

13. **How can you create your own events?**
=>Creating a Custom Event: To create a custom event we **use the Event constructor or CustomEvent interface**. The Event constructor creates an Event and CustomEvent creates an Event with more functionality. The below steps are followed in order to create one using a new Event. We create an event using the Event constructor

14. **What are clusters?**
=>: **a number of similar things growing or grouped closely together** : bunch a cluster of houses a flower cluster. cluster. verb. clustered; clustering.

15. **How does your Node.js application handle scale? Elaborate**
=>1 — Cloning. The easiest thing to do to scale a big application is to clone it multiple times and have each cloned instance handle part of the workload (with a load balancer, for example). ...

1. 2 — Decomposing. ...
2. 3 — Splitting.

16. **What is the difference between readFile and readFileSync?**
=> **readFileSync() is synchronous and blocks execution until finished. These return their results as return values. readFile() are asynchronous and return immediately while they function in the background**. You pass a callback function which gets called when they finish

17. **What are CORS? How do you configure them? Why do you need them?**
=>Cross-Origin Resource Sharing (CORS) is **an HTTP-header based mechanism that allows a server to indicate any origins (domain, scheme, or port) other than its own from which a browser should permit loading resources**

18. What is rate limiting?
=> Rate limiting is a strategy for limiting network traffic. It puts a cap on how often someone can repeat an action within a certain timeframe – for instance, trying to log in to an account.

19. How does middlewares work in express?
=> Middleware functions are functions that have access to the request object (req), the response object (res), and the next function in the application's request-response cycle.

20. What is the difference between Encryption and Hashing?
=> Since encryption is two-way, the data can be decrypted so it is readable again. Hashing, on the other hand, is one-way, meaning the plaintext is scrambled into a unique digest, through the use of a salt, that cannot be decrypted.

21. What is the difference between https and http?
=> HTTPS is HTTP with encryption. The only difference between the two protocols is that HTTPS uses TLS (SSL) to encrypt normal HTTP requests and responses. As a result, HTTPS is far more secure than HTTP. A website that uses HTTP has http:// in its URL, while a website that uses HTTPS has https://.

22. What is TLS?
=>TLS Basics. Transport Layer Security (TLS) encrypts data sent over the Internet to ensure that eavesdroppers and hackers are unable to see what you transmit which is particularly useful for private and sensitive information such as passwords, credit card numbers, and personal correspondence

23. What is AES?
=> The AES engine encrypts the plain text (source data) into cipher text (encrypted data) and sends it to the NAND flash for storage

24. What is JWT Token? Why do we need to use JWT? What are some pros and cons?
=> A JWT is a mechanism to verify the owner of some JSON data. It's an encoded, URL-safe string that can contain an unlimited amount of data (unlike a cookie) and is cryptographically signed. When a server receives a JWT, it can guarantee the data it contains can be trusted because it's signed by the source

25. **What is the difference between authorisation and Authentication?**
=>Simply put, authentication is the process of verifying who someone is, whereas authorization is the process of verifying what specific applications, files, and data a user has access to

26. **What is the difference between JS on the browser and node?**
=>Unlike the browser where Javascript is sandboxed for your safety, node. js has full access to the system like any other native application

27. **What is V8?**
=>V8 is the name of the JavaScript engine that powers Google Chrome. It's the thing that takes our JavaScript and executes it while browsing with Chrome. V8 provides the runtime environment in which JavaScript executes