**1) What is Node Js and how is It used?**  
Node JS is server-side Java scripting that is based on Google’s V8 script . Its main motive is to build scalable web applications which are simple and accessible. Node JS has proven to be effective even in building applications used for video streaming, gaming, IoT devices and apps and for developing real-time web applications, generic applications and network-based applications. It is mainly for event driven and non- blocking servers.

**2) What are some of the features of Node JS?**  
Being tied in a single thread, Node JS is highly scalable and makes use of Java as its scripting language. It also puts to use Asynchronous which is the event-driven I/O. This enables the single thread access rather than diverse and different threads. Even with the single thread it can achieve high performance.

**3) What sort of applications can be built with Node JS application?**  
Whether you are a startup or a large scale enterprise, you can build just about anything with Node JS application. For multiple I/O operations and other real-time systems Node.js excels Compared to other frameworks Node.JS performs well due to its event-driven nature.

Types of applications you can build with Node JS-

* + Social Media applications
  + Project management applications
  + Discussion platforms
  + Live stream video applications
  + Gaming applications
  + IoT applications and devices

**4) Why is Node.js single threaded?**  
It is single threaded because for async processing. For the usual web loads, due to this feature the performance enriches which is considered much better than the multi-thread applications.  
 **5) Can Node JS be used for Machine Learning?**  
For machine learning Node.js might not be the best tool but for other application Node JS is the best fit. They are not very good at heavy-task computation like Machine learning but instead they are preferred for real-time based web applications. For Machine learning and AI based elements integration we would suggest to go with Python.

**6) Which database to use for Node JS?**  
On the lines of several hosting providers Node.js can function with any database like MySQL, MongoDB etc.

* 1. **What are Node.js modules?**  
     Node.js modules are merely libraries and chunks of code which the developers use in order to complete their work rapidly. It is due to these modules that Node.JS is a wonderful choice. The Node.js also has an active community. The number of modules on the primary Node.JS registry has been increasing by the way. Hire any competent Node.js developer and you will see that any issue can be resourced effectively with the help of this active community.

**7) What is the meaning of event-driven Node.js?**  
The meaning of event-driven is that there are a few events dispatched when any given action happens. Such events can also be used to perform an equivalent action. The event-driven feature comes from the asynchronous feature of Node.js. There is also an Event Emitter in the core of NodeJS which takes the role of dispatching and receiving messages. If we want to watch the condition of reading a file then Node.js is useful.

**8) How can I scale Node.js application?**  
The Node JS application can be scaled horizontally and vertically. It also implies that you have to run on multiple occurrences of your app inside one machine and spread the traffic across various CPU threads. Node.JS also has a module called cluster which is extremely helpful. In the case of horizontal scaling it works similar to any other routine back-end application.

**9) How does the event loop work?**  
Event loop is a mechanism that waits and dispatches events and messages in a program. Here the event loops are prime middle flow constructs. This means that every time a request is about to be taken care of, it is put on the event loop and processed when it is ready to do so. So, node instead of completing the task on its own delegates the task of handling the overall system. This is effectively useful as Node.Js can in the meantime handle other tasks till the time it is waiting for this task to finish. The presence of Event loop makes it faster and more effective as compared to other technologies.

**10) What is callback hell in Node Js?**  
When there are heavily nested callbacks Callback hell can be one of the consequences. These heavily nested callbacks make the code indistinguishable and makes it difficult to maintain.

**11) How can these callbacks be fixed?**  
There are majorly three ways with which callbacks can be fixed. They are:

* + Every single error be taken care of separately
  + The code has to be kept narrow
  + Split the callbacks into minute and independent functionalities which can be called with parameters. These can then be joined back in order to get one desired functionality.

**12) What is the role of REPL in Node Js?**  
The Read Eval Print loop as the name itself suggests is used for Reading, Evaluating, Printing and Looping. The REPL in Node JS is used for ad-hoc JavaScript frameworks which allows entry to the JavaScript directly in to the shell prompt and analyze the result thereafter. REPL ‘s role is extremely critical when it comes to testing, debugging, and even experimenting.

**13) What are the types of API functions in Node JS?**  
Node JS comes with two main types of functionality namely Blocking function and Non-Blocking functions.

**Block functions-**In this type of functionality all the other code is blocked from execution till the point an I/O event happens which hitherto was not happening. These blocking functions are accomplished synchronously.

Take the case of one example which will strengthen your understanding on the same

const fs = require('fs');

const data = fs.readFileSync('/file.md'); // blocks here until file is read

console.log(data);

// moreWork(); will run after console.log

If you observe the code, the second line of the code blocks the execution of Javascript as observed in the second line. This blocking takes place till the full file is read.

**Non-Blocking functions**

In this case Multiple I/O can be performed with the stopping of the program. These functions also can be accomplished asynchronously. We will throw a code to make stronger your understanding on the same.

const fs = require('fs');

readFile ('/file.md', (err, data) => {

if (err) throw err;

console.log(data);

});

// moreWork (); will run before console.log

As fs. readFile file is a multi-functional non-blocking one, more work () does not exactly have to delay for the file to read to complete before it is called. Due to this feature the throughput so achieved is on a higher scale.

**14) What are the functionalities of NPM in Node JS?**  
Node Package Manager or NPM basically has two functions:

It proves as an online storehouse for Node JS packages

It also contains a command In-line utility feature that helps in installation of packages and dependency management of different Node JS packages.

**15) What are the different streams in Node JS?**  
Streams are chiefly those objects that allow the reading of data from the source and also writing of the data onto the destination.  There are four main types of streams as listed below:

* + **Readable:** this helps in reading operation
  + **Writable:** this helps in the writing operation
  + **Duplex:** this helps in both reading and writing operation
  + **Transform:** This is a form of duplex operation that undertakes computations considering the available input.

**16) What are the benefits of using Node JS?**  
If you are looking at easily scalable network programs then Node JS is the answer. Some of its chief advantages are

* + It is really fast
  + It usually doesn’t block
  + It has a unified programming language and data type
  + Asynchronous makes it easy
  + Code sharing and reuse
  + Availability of many free tools
  + It has great concurrency

**17) What is the difference between Angular JS and Node JS?**While Node JS is a run-time system, AngularJS is a web application development.