Both can be used to switch to an asynchronous mode of operation by listener functions.

process.nextTick() sets the callback to execute but setImmediate pushes the callback in the queue to be executed. So the event loop runs in the following manner

timers–>pending callbacks–>idle,prepare–>connections(poll,data,etc)–>check–>close callbacks

In this process.nextTick() method adds the callback function to the start of the next event queue and setImmediate() method to place the function in the check phase of the next event queue.

20. How does Node.js overcome the problem of blocking of I/O operations?

Since the node has an event loop that can be used to handle all the I/O operations in an asynchronous manner without blocking the main function.

So for example, if some network call needs to happen it will be scheduled in the event loop instead of the main thread(single thread). And if there are multiple such I/O calls each one will be queued accordingly to be executed separately(other than the main thread).

Thus even though we have single-threaded JS, I/O ops are handled in a nonblocking way.

21. How can we use async await in node.js?

Here is an example of using async-await pattern:

Contents

Beginner Node.js Interview Questions

Intermediate Node.js Interview Questions

Let's get Started

Premise

> "Any application that can be written in JavaScript, will eventually be written in JavaScript." -Jeff Atwood

This was said back in 2007, and we can say that it is proving true till now. You can think of any technical keyword and there might be a JavaScript library build around it. So if it's so popular and in demand, this can be a great programming language to learn. But that's not the only skill that is required, since you have to apply this to solve practical problems. And one of such problems is to build scalable products.

Gen Z backend

After jQuery animation dev shifted to a single page application for better control of ui/ux and thus came frontend frameworks such as angular js and angular. After that JavaScript was made available to port into literally any modern machine that exists and runs as a standalone application i.e Node.js. It was widely accepted as a backend framework and comes to the top, 2nd year in a row in 2020 of StackOverflow survey.

As developers are busy getting an experience in node.js it's nice to have a curated list of **Node.js interview questions** to revise. Also, to further consolidate your knowledge on Javascript, refer to this source.

Beginner Node.js Interview Questions

1. What is a first class function in Javascript?

Advanced Node.js Interview Questions

30. What is an Event Emitter in Node.js?

EventEmitter is a Node.js class that includes all the objects that are basically capable of emitting events. This can be done by attaching named events that are emitted by the object using an eventEmitter.on() function. Thus whenever this object throws an even the attached functions are invoked synchronously.

```
const EventEmitter = require('events');
class MyEmitter extends EventEmitter {}
const myEmitter = new MyEmitter();
myEmitter.on('event', () => {
 console.log('an event occurred!');
});
myEmitter.emit('event');
```

31. Enhancing Node.js performance through clustering.

Node.js applications run on a single processor, which means that by default they don’t take advantage of a multiple-core system. Cluster mode is used to start up multiple node.js processes thereby having multiple instances of the event loop. When we start using cluster in a nodejs app behind the scene multiple node.js processes are created but there is also a parent process called the **cluster manager** which is responsible for monitoring the health of the individual instances of our application.