Topic 5 Node.js and Express.js

Interview Questions and Answers

Q1. What is node.js?

Ans:

- Node.js is a powerful JavaScript run time environment developed on Google's open-source chrome's V8 engine.
- Node.js run time environment contains everything that is needed to execute JavaScript code.
- It compiles JavaScript code directly into the native machine.
- It enables the developer to run JavaScript code directly code in a computer instead of in a browser.

Q2. Why use node.js?

Ans: Following are the features of the node.js that make us choose node.js among any other technologies for the backend:

- **Asynchronous and Event-Driven:** All APIs of node.js are non-blocking, which is asynchronous. This means that the node.js based server does not wait for the response of an API but simply continues with the next API request, and the Event loop will help in returning the data when the previous API returned the response.
- **Fast:** Being developed on Google's open-source chrome v8 engine. Node.js is very faster.
- **Single Threaded:** Node.js used a single-threaded model with event

looping. The mechanism of event looping and event queue will help node.js to respond asynchronously.

- **Cost-effective:** it can handle multiple requests with a single thread which reduces the cost of servers.
- **No Buffering:** node.js application never buffers any data. It will return responses in chunks of data.

Q3. Explain base routes in Node.js?

Ans:

- It is a prefix added to all the routes
- It helps in versioning of project
- Using base routes, we can add a prefix to all our routing methods

Q4. What are the advantages of using promises instead of callbacks?

Ans: Both callbacks and promises are used to handle the asynchronous operations, callback callbacks can create callback hell when using multiple asynchronous operations. Promises provide a better way of handling asynchronous operations over callback. Following are the advantages of using promise over callbacks:

- Improves code readability
- Better handling of the asynchronous operation
- Better flow of Control definition in asynchronous logic
- Better error handling

Q5. Explain Event Queue and Event loop in Node.js

Ans:

Event Queue: event queue is used to collect all the requests store them in it.

Event Loop: event loop check for any request placed in the event queue and process it. If no requests are found, then it waits for any incoming requests.

Q6. What is NPM?

Ans:

- NPM stands for the node package manager, it is the default package, open-source manager JavaScript run time environment Node.js.
- NPM is a registry of reusable modules and packages developed by other packages.
- Anyone can use packages and modules from NPM to achieve specific functionally.
- One can use the packages of npm and can also publish packages or modules to the NPM library.

Q7. What are is a module and what are the different types of modules in Node.js?

Ans:

The module is a functionality that is organized in one or more JavaScript files and can be reused throughout the node.js application.

Following are three types of modules in node.js:

Core Modules: these are the modules, which are in-built, and you can use without installation

User-defined modules: these are the modules, which are created by you, which you want to reuse in the rest of the application.

Node Package Manager: these include a group of modules or packages, developed by other developers, which can be used in your application by installing them.

Q8. What is Backend development?

Ans:

- The backend runs on the server and has no direct interaction with the user.
- Creates a connection between web and database.
- Client-side application.
- The main purpose of the backend is to listen to the request sent by the frontend application, process the request, and respond with appropriate content.

Q9. What is the single-threaded model?

Ans:

- Makes use of only one thread and there is no need to create a new thread, because of the event loop.
- The request never needs to wait before processing.
- Uses queue to store requests and then process them.
- Can never result in deadlock or race conditions.
- Utilizes less memory where there is an increase in current requests.

Q10. What is the multi-threaded model?

Ans:

- Maintains a limited thread pool and a new thread is assigned to different requests.
- If the thread limit is exceeded with more concurrent requests, the request might need to process the request.
- Use the incoming-request model.
- Can result in deadlock or race conditions
- Utilized more memory when there is an increase in concurrent requests

Q11. How do you differentiate between backend and frontend?

Ans:

Following are the difference between frontend and backend:

Frontend development	Backend development
Refers to the application that the user interacts with	Creates a connection between website and database which is used to request and fetch content
Client-side application	Server-side application
Has direct interaction with the user	Has no direct interaction with the user
Technologies used include HTML, CSS, and JavaScript	Technologies used include PHP, Node.js, JAVA, and .NET

Q12. What is the difference between a single-threaded model and a multi-threaded model?

Ans: Following are the major differences between single threaded model and multi-threaded model:

Multi-Threaded	Single-Threaded
Maintains a limited thread pool and a new request is assigned to a different request	Makes use of only one thread and there is no need to create new threads, because of the event loop.
If the thread limit exceeded with more concurrent requests, the request might need to wait to process the request	The request never needs to wait before processing
Uses incoming-request model	Uses queue to store the requests
Can result in deadlock or race conditions	Can never result in deadlock or race conditions
Utilized more memory when there is an increase in concurrent requests	Utilizes less when there is an increase in concurrent requests