Node.js MCQ Test Questions

Below are 50 multiple-choice questions (MCQs) based on Node.js fundamentals, covering topics such as modules, file system operations, HTTP servers, events, and basic concepts typically found in an introductory Node.js repository. The questions are divided into theory-based (1-25) and code-based (26-50) sections.

Theory-Based Questions

1. **What is Node.js primarily built on?** A) Java Virtual Machine B) V8 JavaScript Engine C) Python Interpreter D) Ruby Runtime **Answer: B**

2. **Which module is used for handling file system operations in Node.js?** A) http B) fs C) path D) os **Answer: B**

3. **What does the 'require()' function do in Node.js?** A) Starts an HTTP server B) Imports modules or files C) Handles events D) Reads directories **Answer: B**

4. **Node.js is described as single-threaded. What does this mean for its execution model?** A) It can only run one CPU core B) It uses non-blocking I/O for concurrency C) It blocks on every operation D) It requires multiple processes by default **Answer: B**

5. **Which method is synchronous for reading a file in the fs module?** A) fs.readFile() B) fs.readFileSync() C) fs.appendFile() D) fs.writeFile() **Answer: B**

6. **What is the purpose of the 'path' module in Node.js?** A) To handle HTTP paths B) To work with file paths and directories C) To parse JSON D) To create events **Answer: B**

7. **In Node.js, modules are loaded using which keyword?** A) import B) require C) include D) load **Answer: B**

8. **What is the default export mechanism in Node.js CommonJS modules?** A) module.exports B) exports.default C) return statement only D) global object **Answer: A**

9. **Which event is emitted when a file is opened successfully using fs.open()?** A) 'error' B) 'open' C) 'read' D) 'close' **Answer: B**

10. **What does process.cwd() return?** A) The current working directory B) The process ID C) The environment variables D) The current user **Answer: A**

11. **In Node.js, what is the role of the EventEmitter class?** A) To handle HTTP requests B) To emit and listen for custom events C) To read files asynchronously D) To parse command-line arguments **Answer: B**

12. **Which module is core to creating an HTTP server in Node.js?** A) fs B) http C) url D) querystring **Answer: B**

13. **What is the typical port used for a basic HTTP server example in Node.js?** A) 3000 B) 8080 C) 80 D) 5000 **Answer: A**

14. **Node.js uses which I/O model to handle multiple operations efficiently?** A) Blocking I/O B) Non-blocking I/O with callbacks C) Thread-based I/O D) Synchronous I/O **Answer: B**

15. **What does fs.existsSync() return?** A) A boolean indicating if a path exists B) The file contents C) The file stats D) An error if not found **Answer: A**

16. **In module creation, what is assigned to module.exports to make a function available externally?** A) The function name B) An object with the function C) A string D) Nothing, it's automatic **Answer: B**

17. **What is the purpose of process.argv in Node.js?** A) To access command-line arguments B) To exit the process C) To set environment variables D) To log messages **Answer: A**

18. **Which method creates a directory synchronously?** A) fs.mkdirSync() B) fs.mkdir() C) fs.createDir() D) path.mkdir() **Answer: A**

19. **Node.js callbacks often follow which error-handling pattern?** A) First parameter is error, second is data B) First parameter is data, second is error C) Errors are thrown only D) No error parameter needed **Answer: A**

20. **What does the 'os' module provide information about?** A) Operating system details like CPU and memory B) File paths C) HTTP status codes D) Event types **Answer: A**

21. **In an HTTP server, what event is listened for to handle incoming requests?** A) 'connection' B) 'request' C) 'data' D) 'end' **Answer: B**

22. **What is path.join() used for?** A) To concatenate paths in a platform-independent way B) To join strings C) To resolve URLs D) To split paths **Answer: A**

23. **Node.js global objects do not need to be required. Which one is used for console output?** A) console B) log C) print D) output **Answer: A**

24. **What happens if you call require() on the same module twice?** A) It reloads every time B) It caches and returns the same instance C) It throws an error D) It creates a new copy **Answer: B**

25. **Which function is used to exit a Node.js process?** A) process.exit() B) system.exit() C) end() D) stop() **Answer: A**

Code-Based Questions

26. **Consider the code:** `const fs = require('fs'); const data = fs.readFileSync('file.txt', 'utf8'); console.log(data);` **What does this do?** A) Reads file asynchronously B) Reads file synchronously and logs content C) Writes to file D) Checks if file exists **Answer: B**

27. **In this code:** `module.exports = { add: (a, b) => a + b };` **How do you use it in another file?** A) const mod = require('./file'); mod.add(1,2); B) import mod from './file'; C) load('./file'); D) include mod; **Answer: A**

28. **Code:** `const http = require('http'); const server = http.createServer((req, res) => { res.end('Hello'); }); server.listen(3000);` **What happens when accessed at localhost:3000?** A) Errors out B) Displays 'Hello' C) Infinite loop D) Nothing, no event listener **Answer: B**

29. **What is the output of:** `const path = require('path'); console.log(path.join('/a', 'b', 'c'));` **(on Unix-like system)?** A) /a/b/c B) /a b c C) ab/c D) Error **Answer: A**

30. **Code:** `const EventEmitter = require('events'); const emitter = new EventEmitter(); emitter.on('greet', () => console.log('Hi')); emitter.emit('greet');` **What is logged?** A) Nothing B) 'Hi' C) 'greet' D) Error **Answer: B**

31. **In:** `fs.readFile('file.txt', (err, data) => { if (err) console.log('Error'); else console.log(data); });` **What is the first callback parameter?** A) Data B) Error object or null C) Boolean D) File path **Answer: B**

32. Code: `console.log(process.argv[2]);` If run as `node script.js hello` what is output? A) node B) script.js C) hello D) undefined **Answer: C**

33. What does `fs.writeFileSync('file.txt', 'content');` do? A) Appends content B) Overwrites or creates file with 'content' C) Reads the file D) Deletes the file **Answer: B**

34. Code: `const os = require('os'); console.log(os.cpus().length);` What does it output? A) Number of CPUs B) OS name C) Memory usage D) Uptime **Answer: A**

35. In module: `function greet() { return 'Hello'; } module.exports = greet;` Usage: `const g = require('./mod'); console.log(g());` Output? A) function greet() B) 'Hello' C) undefined D) Error **Answer: B**

36. Code: `fs.mkdirSync('newdir');` What occurs if 'newdir' exists? A) Ignores B) Throws error C) Overwrites D) Renames **Answer: B**

37. What is the output: `path.resolve('a', '../b');` (assuming current dir /home)? A) /home/b B) a/../b C) /home/a/b D) Error **Answer: A**

38. Code: `server.on('request', (req, res) => { res.writeHead(200); res.end(); });` What HTTP status is set? A) 404 B) 200 C) 500 D) None **Answer: B**

39. In: `process.on('exit', () => console.log('Exiting')); process.exit(0);` What is logged? A) 'Exiting' B) 0 C) Nothing D) Error **Answer: A**

40. Code: `const data = fs.readFileSync('nonexistent.txt');` What happens? A) Returns empty string B) Throws ENOENT error C) Waits indefinitely D) Logs warning **Answer: B**

41. What does `emitter.emit('event', 'data'); emitter.on('event', (arg) => console.log(arg));` log? (Note: order matters) A) 'data' B) Nothing (listener after emit) C) 'event' D) Error **Answer: B**

42. Code: `module.exports.name = 'Node';` In another file: `const m = require('./mod'); console.log(m.name);` Output? A) undefined B) 'Node' C) module D) Error **Answer: B**

43. In HTTP: `res.write('Hi'); res.end(' World');` What is sent to client? A) 'Hi World' B) 'Hi' only C) Error D) ' World' only **Answer: A**

44. Code: `fs.appendFileSync('file.txt', 'add');` What does it do? A) Overwrites B) Appends 'add' to end C) Reads D) Creates if not exists and appends **Answer: D**

45. What is output: `console.log(__dirname);` ? A) Current directory of the script B) Root directory C) User home D) Temp directory **Answer: A**

46. Code: `http.createServer().listen(3000, 'localhost');` What binds the server? A) All interfaces B) Only localhost C) Port only D) Error **Answer: B**

47. In: `fs.readdirSync('.');` What is returned? A) Array of files/directories in current dir B) Single file C) Path string D) Stats object **Answer: A**

48. What is process.env.NODE_ENV default? A) 'development' B) undefined C) 'production' D) 'test' **Answer: B**

49. What happens in: `server.close();` for an HTTP server? A) Stops accepting new connections B) Crashes process C) Restarts D) Logs error **Answer: A**

50. Code: `path.extname('file.js');` Output? A) '.js' B) 'js' C) 'file' D) Error **Answer: A**