Backend Questions

1. Explain the difference between frontend and backend development?
2. What is the difference between JavaScript and Node.js?
3. What is the difference between asynchronous and synchronous functions?
4. What is NodeJS? Explain in detail the working of NodeJS.
5. What is NPM?
6. Explain CommonJS vs ModuleJS syntax in NodeJS with examples.
7. What is the package.json file?
8. Explain Event Loop in Node.js?
9. How do you install, update, and delete a dependency(global, local, and dev)?
10. How do you manage packages in your Node.Js project?
11. How do you create a simple server in Node.js that returns Hello World?
12. What is Express and why use it?
13. How do you create a simple Express.js application?
14. What is callback hell? How do we overcome it?
15. What is the purpose of an API (Application Programming Interface) in a backend application?
16. Explain the concept of routing and how it is implemented in backend frameworks.
17. Explain the concept of middlewares in Node/Express.
18. What are the different types of HTTP requests?
19. Explain about different HTTP status codes in detail.
20. Difference between SQL and NoSQL databases.
21. What is MongoDB and its advantages and disadvantages?
22. How would you connect a MongoDB database to Node.js?
23. What is mongoose and why use it?
24. What is RDBMS? How is it different from DBMS?
25. What are Constraints in SQL?
26. What is a Primary Key, Foreign Key and difference between them?
27. What is a Join? List its different types.
28. What is an Index? Explain its different types.
29. What is a Query?
30. List the different types of relationships in SQL.
31. What is Normalization and Denormalization?
32. What are TRUNCATE, DELETE, and DROP statements and differences between them?
33. How do you handle error and exception handling in node/express application?
34. How do you handle input validation and data sanitization in a backend application?

35. How do you handle cross-origin resource sharing (CORS) in a backend application?
36. What are the key considerations when designing a RESTful API?
37. What are the differences between stateless and stateful communication in a backend system?
38. How do you handle versioning in a backend API?
39. What is the purpose of rate limiting and the process of implementing rate limiting to prevent abuse or excessive API usage.
40. What is the role of web sockets in real-time communication in a backend application?
41. How does caching improve the performance of a backend application?
42. Describe the process of implementing a caching strategy for a backend application.
43. How do you handle database transactions in a backend application?
44. Explain the concept of data sharding and its benefits in scaling a backend database.
45. What is the role of indexing in a database and how does it impact performance?
46. Describe the process of authentication and authorization in a backend application.
47. How do you ensure the security of sensitive data in a backend system?
48. What are worker threads in NodeJS?
49. Explain the concept of containerization and its benefits in backend deployment.
50. How do you ensure high availability and fault tolerance in a backend system?
51. What is the role of a reverse proxy in backend infrastructure?
52. Describe the process of scaling a backend application horizontally and vertically.
53. How do you handle long-running tasks in a backend system?
54. Explain clustering in NodeJS and how do we achieve it?
55. Explain the concept of Access Token, Refresh Token.
56. Explain the concept of serverless computing and its benefits in backend development.
57. What are the key considerations for securing a backend application against common vulnerabilities?
58. Explain the concept of event-driven architecture and its use in backend systems.
59. What are the benefits of using microservices architecture in backend development?
60. What is the role of a service mesh in microservices architecture?
61. Describe the role of a load balancer in a distributed backend system.
62. Explain the concept of message queues and their significance in backend architecture.
63. Explain the concept of eventual consistency in distributed databases.

64. What are the best practices for logging and error handling in a backend application?
65. Describe the process of designing and implementing a task scheduling system.
66. How do you ensure data integrity and prevent data corruption in a backend system?