

# 100 MEAN Stack Interview & Practice Questions

1. What is MEAN Stack?
2. Explain the role of each technology in the MEAN stack.
3. What are the advantages of using MEAN over other stacks?
4. What is MongoDB, and why is it used in MEAN?
5. How does MongoDB store data?
6. What is the difference between SQL and NoSQL databases?
7. Explain CRUD operations in MongoDB.
8. What is Mongoose, and why do we use it with MongoDB?
9. How do you define a schema in Mongoose?
10. What are middleware functions in Mongoose?
11. What is Express.js, and why is it used in MEAN?
12. How do you create a basic Express.js server?
13. What is middleware in Express.js?
14. What are the different types of middleware in Express.js?
15. How do you handle routes in Express.js?
16. How do you connect MongoDB with Express.js?
17. What is CORS, and why is it important?
18. How do you handle errors in Express.js?
19. What is Angular, and why is it used in MEAN?
20. What is TypeScript, and why does Angular use it?
21. What are components in Angular?
22. What is the difference between a component and a module in Angular?
23. What are Angular services, and how do they work?
24. What are Angular directives?
25. Explain the difference between structural and attribute directives.
26. What is Angular routing, and how does it work?
27. How do you make HTTP requests in Angular?
28. What is dependency injection in Angular?

29. What are Angular lifecycle hooks?
30. What is Angular CLI, and how is it used?
31. How do you implement form validation in Angular?
32. What is RxJS, and why is it used in Angular?
33. What are observables and promises in Angular?
34. What is state management in Angular?
35. What is NgRx, and how does it work?
36. What is Node.js, and why is it used in MEAN?
37. What is the event loop in Node.js?
38. What are streams in Node.js?
39. What are modules in Node.js?
40. What is npm, and how is it used?
41. What is package.json in Node.js?
42. How do you use environment variables in Node.js?
43. What is authentication, and how do you implement it in MEAN?
44. What is JWT (JSON Web Token)?
45. How do you handle user sessions in MEAN?
46. What is bcrypt, and how is it used for password hashing?
47. How do you implement social login in a MEAN app?
48. What is REST API?
49. What is GraphQL, and how does it compare with REST?
50. How do you test APIs in MEAN?
51. What is Postman, and how is it used for API testing?
52. What is the difference between GET, POST, PUT, and DELETE?
53. How do you handle file uploads in a MEAN app?
54. What is Multer, and how does it work?
55. How do you secure an Angular application?
56. What is lazy loading in Angular?
57. What is a Progressive Web App (PWA)?
58. How do you optimize performance in a MEAN application?

59. What are some security best practices for MEAN apps?
60. What is rate limiting in Express.js?
61. How do you implement caching in a MEAN app?
62. What is Helmet in Express.js?
63. How do you handle image uploads in MEAN?
64. How do you schedule tasks in Node.js?
65. What is the purpose of Angular pipes?
66. How do you implement Angular animations?
67. What is service worker in Angular?
68. How does Angular handle two-way data binding?
69. How do you configure MongoDB indexes for better performance?
70. What is Angular Universal, and why is it used?
71. What are Angular guards, and how do they work?
72. What is a singleton service in Angular?
73. How do you configure production builds in Angular?
74. How do you monitor and log errors in a MEAN app?
75. What is the role of WebSockets in MEAN?
76. How do you integrate third-party APIs in MEAN?
77. What is an API Gateway?
78. How do you optimize SEO in a MEAN application?
79. How does server-side rendering (SSR) work in Angular?
80. How does Angular handle forms (Reactive Forms vs Template-Driven Forms)?
81. What is a module federation in Angular?
82. What is Docker, and how can it be used with MEAN?
83. How does TypeScript improve Angular development?
84. How do you deploy a MEAN application?
85. What is CI/CD, and how does it apply to MEAN?
86. How do you scale a MEAN application?
87. What are microservices, and how do they relate to MEAN?
88. How do you handle pagination in MongoDB?

89. What is load balancing?
90. How do you implement Redis caching in a MEAN application?
91. What is Nodemon, and why is it useful?
92. How do you integrate payment gateways in a MEAN app?
93. What is Next.js, and does it work with MEAN?
94. How do you prevent Cross-Site Scripting (XSS) in Angular?
95. How does Angular handle HTTP interceptors?
96. What is the difference between client-side and server-side rendering?
97. What is an event emitter in Node.js?
98. How do you handle database migrations in MongoDB?
99. How do you configure Angular services as singletons?
100. What is the difference between ngIf and hidden attributes in Angular?