

TOP 12

MERN STACK INTERVIEW QUESTIONS

@Ajaytripathi





Q1. What is the MERN Stack?

MERN stands for MongoDB, Express.js, React.js, and Node.js. It's a full-stack JavaScript framework for building end-to-end web applications.

- MongoDB – NoSQL Database
- Express.js – Backend framework
- React.js – Frontend library
- Node.js – Runtime environment

Q2. How does the frontend (React) communicate with the backend (Express/Node)?

Using HTTP requests (mostly `fetch()` or `axios`) to interact with APIs created in Express.js.

The flow:

React → Axios/fetch → Express Route → MongoDB → Response → React UI

Comment **“MERN”** to get the complete guide





Q3. What is Mongoose and why is it used?

Mongoose is an ODM (Object Data Modeling) library for MongoDB. Features -

- Schema-based validation
- Middleware hooks
- Easy CRUD operations

Example - `const User = mongoose.model('User', userSchema);`

Q4. What are some performance optimization techniques in MERN apps?

Frontend:

- Lazy Loading Components
- Memoization (React.memo, useMemo)

Backend:

- Caching (Redis)
- DB Indexing
- Pagination
- Load balancing

Comment “MERN” to get the complete guide





Q5. How does React handle reconciliation and virtual DOM updates?

React uses a Virtual DOM and a diffing algorithm to detect changes.

It compares the new virtual DOM with the previous one and updates only the changed nodes using a batching strategy, improving performance.

Q6. How do you implement secure authentication in a MERN app?



- Use bcrypt to hash passwords
- Use JWT for access tokens
- Store tokens in httpOnly cookies to prevent XSS
- Use middleware in Express to verify tokens

Bonus: Add refresh tokens for better security.



Comment **"MERN"** to get the complete guide



Q7. What's the difference between SQL and MongoDB data modeling?

SQL: Relational, normalized schemas, joins

MongoDB: Document-based, denormalized for fast reads

Use embedded documents in MongoDB where possible

Use references only for large, reusable datasets

Q8. What is the event loop in Node.js and how does it work?

Node.js is single-threaded and uses the event loop to handle async operations. Tasks like file I/O, DB access are sent to the event queue, and callbacks are executed non-blockingly when ready.



Comment **"MERN"** to get the complete guide



Q9. How to write unit and integration tests in a MERN stack?

- Frontend (React): Jest + React Testing Library
- Backend (Node/Express): Mocha, Chai, Supertest
- Use mocking (e.g., Mock Service Worker, Sinon) for API calls

Q10. Explain the full deployment process of a MERN app (production-ready)?

- Backend: Host on Render / Railway / VPS
- Frontend: Deploy React on Netlify/Vercel
- MongoDB: Use MongoDB Atlas
- Env Vars: Secure via .env
- Reverse Proxy + HTTPS: Use Nginx + SSL
- Optional: Dockerize + CI/CD (GitHub Actions)



Comment “MERN” to get the complete guide



Q11. What is CORS and how do you handle it in a MERN app?

CORS (Cross-Origin Resource Sharing) prevents requests between different origins.

To fix:

```
const cors = require('cors');  
app.use(cors({ origin: 'http://localhost:3000', credentials: true }));
```

Q12. What is Redux and when should you use it?

Redux is a state management tool used when:

- You have deeply nested components sharing data
- You want predictable state changes
- You want to use middleware like Redux Thunk



Comment **“MERN”** to get the complete guide



Want more like this ?

Get the complete MERN Full Stack Web Development Kit

- **250+** Frontend, **50+** Backend & **20+** Full-Stack Projects with complete source code
- **700+** Interview Questions covering top tech companies
- Hidden **Web Dev Tools** + **Web 3.0** & MERN **Roadmaps**
- Complete tutorials –
 - HTML, CSS, JavaScript
 - React & Angular
 - Node.js & Express.js
 - MongoDB
 - Bootstrap
 - **BONUS:** Python Django



Follow Telegram channel :
TechTalks_with_Ajay



@TECHTALKS_WITH_AJAY