

Q1. What is Eureka and what primary problem does it solve in the Microservices space?

Answer:

Q2. Explain the core architecture or components of Eureka and how they work together.

Answer:

Q3. What are the main alternatives to Eureka and when would you choose it over them?

Answer:

Q4. How do you install, configure, and get started with Eureka for a new project?

Answer:

Q5. What are the key configuration options or settings that affect Eureka's behavior in production?

Answer:

Q6. How does Eureka handle scalability, reliability, and high availability?

Answer:

Q7. What observability does Eureka provide: metrics, logs, traces, or health endpoints?

Answer:

Q8. What are the common performance bottlenecks in Eureka and how do you address them?

Answer:

Q9. What security best practices should you follow when deploying Eureka?

Answer:

Q10. What are the most common pitfalls or mistakes teams make when adopting Eureka?

Answer:
