SAIREDDY-DOTNETFS

Are you preparing for Java Full Stack interivew

Core Java & OOP

1. Explain OOP principles (Encapsulation, Inheritance, Polymorphism, Abstraction) with examples.
2. Difference between Abstract Class vs Interface (Java 8+ features: default & static methods).
3. How does HashMap work internally? (hashCode, equals, collisions, load factor, resizing).
4. Difference between String, StringBuilder, StringBuffer.
5. What is the difference between Checked, Unchecked, and Error in exceptions?
6. Explain Garbage Collection (GC) in Java. How do you troubleshoot Memory leaks?
7. Difference between final, finally, and finalize().
8. What are functional interfaces? Give examples (Predicate, Function, Consumer).
9. Explain Streams API and give examples of filtering, mapping, and reducing.
10. Multithreading in Java – Difference between Thread, Runnable, Callable, Future, CompletableFuture.

Spring & Spring Boot

1. What's the difference between Spring MVC and Spring Boot?
2. Explain Dependency Injection (DI) and IoC container in Spring.
3. What are Bean Scopes in Spring?
4. Difference between @Component, @Service, @Repository, @Controller annotations.
5. How does Spring Boot Auto-Configuration work?
6. What is the difference between @RestController vs @Controller?
7. Explain Spring Boot Actuator and monitoring.
8. How to implement Global Exception Handling in Spring Boot?
9. Explain Spring Security Authentication & Authorization (JWT/OAuth2).
10. Difference between JpaRepository, CrudRepository, PagingAndSortingRepository.

Database & ORM (SQL + Hibernate/JPA)

1. Difference between Hibernate and JPA.
2. What are first-level cache & second-level cache in Hibernate?
3. Explain N+1 Select Problem in JPA and how to solve it.
4. How do you implement One-to-Many, Many-to-One, Many-to-Many relationships?
5. Write a query to fetch employees with the second highest salary.
6. Explain ACID properties of a transaction.
7. Difference between INNER JOIN, LEFT JOIN, RIGHT JOIN, CROSS JOIN.
8. What are indexes in DB? Advantages and disadvantages.
9. Difference between Optimistic Locking vs Pessimistic Locking in JPA.
10. How do you write native queries in Spring Data JPA?

Frontend (JavaScript, React/Angular)

1. Difference between var, let, and const in JavaScript.
2. What is closure in JavaScript? Example.
3. Explain Promises vs async/await.
4. Difference between null, undefined, and NaN.
5. In React: Difference between functional components and class components.
6. What are React hooks (useState, useEffect, useContext)?
7. In Angular: Difference between ngOnInit and constructor.
8. What are Directives in Angular (structural vs attribute)?
9. Explain two-way data binding in Angular.
10. How do you implement an interceptor in Angular for adding a JWT token?

System Design & Microservices

1. Explain Monolithic vs Microservices architecture.
2. How do microservices communicate (REST, gRPC, Message Queues)?
3. How do you implement Service Discovery in microservices (Eureka/Consul)?
4. What is API Gateway and why do we need it?
5. How to handle circuit breaker pattern (Resilience4j/Hystrix)?
6. How do you design a system for high availability and scalability?
7. Difference between synchronous vs asynchronous communication.
8. How do you handle distributed transactions in microservices?
9. Explain Docker and Kubernetes basics.
10. CI/CD pipeline – how do you automate builds and deployments?

Coding & Problem-Solving

1. Write code to find duplicates in an array.
2. Write code to reverse a linked list.
3. Write SQL query to get top N employees by salary per department.
4. Write a program to check if a string is a palindrome.
5. Write code to implement producer-consumer problem using threads.

Do you find it helpful?

DO YOU LIKE THE POST?
REPOST IT!

THANKS FOR READING