1. Java
   1. when we use default and static method in interface
   2. Java Memory Model
   3. Cohesion
   4. Singleton Pattern
   5. Try with resource
   6. Parallel Stream works
   7. Map() Vs flatMap()
   8. Use of peak() in Stream API
   9. Internal Implementation of HashMap
   10. HashMap uses red-black tree after crossing threshold
   11. Race condition
   12. Fail Fast vs Fail Safe Iterators
   13. Marker Interface
   14. Functional Interface
   15. Serialization
   16. Externalization
   17. Predicate vs Function
   18. Reflection
   19. @transaction which will rollback if there is any specific exception
   20. Executor Service
   21. Callable Interface
   22. Future Task
   23. Optional API (isPresent, isEmpty, get, orElse, orElseGet)
   24. Multiple Inheritance
   25. How garbage collector works
   26. Different types of memory in heap memory
   27. Checked vs Unchecked Exception
   28. Searching take in hash map
   29. Immutable Class
   30. Spring Buffer and Spring Builder
   31. Can we set null to this keyword
   32. Collision in hash map
   33. Lambda expression create class file or not
   34. Generic
2. Spring
   1. IOC and DI
   2. Beans Lifecycle methods
   3. Bean scope
   4. @Qualifier and @Primary annotation
   5. Cross cutting concerns
   6. Types of autowiring
3. Spring Boot
   1. @SpringBootApplication is combination of other annotation
   2. Interceptors
   3. Controller Advice
   4. Global exception handling
   5. Difference between @Configuration and @Component
   6. @ConditionalOnExpression
4. Hibernate
   1. ORM
   2. Cascading
   3. Many to many relationship
   4. N + 1 problem in hibernate
   5. Save() vs persist()
   6. Get() vs load()
   7. Lasy vs eager loading
   8. How to enable 2nd level cache in hibernate
   9. @Entity vs @Table
   10. Session and Session Factory
5. Microservices
   1. Why we use microservice instead of single service
   2. Communication between two services
   3. why we use array list not linked list in controller response
   4. Circuit Breaker
   5. Spring batch
   6. Disadvantage of microservices
   7. Idempotent
   8. @Projection annotation
   9. @Service vs @Repository
   10. Spring Actuator
6. Angular
   1. Promise Vs Observable
      1. Promise – Handles a single event when an async operation completes or fails
      2. Observable – It is like stream and allows you to pass zero or more events
   2. Impure Vs Pure Pipes
   3. How ngModel works
   4. Subject Vs Observable
   5. RxJS
   6. DOM sanitization
   7. Life cycle methods
   8. Custom pipes
   9. Communication between parent and child
   10. Lazy loading
   11. AOT vs JIT
   12. Types of Directive
7. Design patterns
   1. Design patterns in Java
   2. Design patterns in spring
   3. Design patterns in spring boot
   4. SOLID
   5. Singleton
   6. Builder
   7. Factory method
   8. Decorator
   9. Strategy
   10. CQRS Patterns
   11. Composite
8. System Design
   1. ACID properties
   2. SQL vs No SQL
   3. Kafka vs Rabbit MQ
   4. Distributed transaction (Saga)
   5. API Gateway