

first question:

self-introduction: always try to explain things like first your name, working in company name, total numbers of experience, project name, project technology, overview of the project and roles and responsibility.

From Java:

Which version are you using?

If more than Java 8, like Java 11,

What are the new features introduced in Java 10 and 11?

Java basic:

Explain the oops concept.

What is polymorphism?

Can we override the static method?

Can we change the return type of the overloaded method?

Can we override the protected method? If yes, can we change the access specification from protected to default or public?

Can we override or overload static methods? If not, why, and if yes, how?

What is the use of the hashCode() and equals() methods, and where and how can we create

What are serialisation and deserialization in Java?

What is transient in Java?

collection framework:

Which collection framework did you use in the project?

What is the difference between an array and an arraylist?

What is the difference between hashMap and concurrent hashMap?

What is the purpose of the executor service?

What is the difference between an array list and a linked list?

which data type we can use to maintain insertion order with unique data.

which collection interface and classes you used in your project and for which purpose

Multithreading:

How can we do inter-thread communication?

What is the thread life cycle?

What is the finalise method in Java?

What is the difference between the wait and sleep methods?

Exception:

What is the root class of a Java exception?

What is the hierarchy of Java exceptions?

What are the checked and unchecked exceptions?

Can we use the custom exception as an unchecked exception?

What is try with resource

how we can handle checked and unchecked exception.

String handling:

What is the meaning of immutable? Can we create immutable classes?

What is the difference between StringBuffer and StringBuilder?

What is the difference between String, StringBuffer, and StringBuilder?

If I have two employee objects with the same value and we compare them with the equals method and double equals operator, what will be the output?

Java 8 features:

Explain the Java 8 features.

What are the preferred functional interfaces to introduce?

What is the purpose of introducing default and static methods in Java 8?

What are intermediate and terminal operations in the Java stream API?

What is the difference between a map and a flatmap in Java 8?

What are the covariant return type and type reference?

What is the use of an optional class, and how did you use it in your code to handle null pointer exceptions?

What is the method reference, and what is the difference between the method reference and the lambda expression?

What is the lambda expression?

Why do we need to use lambda expressions when we can provide the implementation by using the anonymous class?

Is it mandatory to use the @FunctionalInterface annotation with a functional interface?

What if we are providing implementation for one interface method in two classes?

Spring boot:

What are the features of the spring boot, or how is the spring boot different from the spring framework?

What is an account, and how can we implement it?

How do we change the port in Spring Boot, or how do I change the default server to something else?

Have you used the spring boot annotation?

@SpringBootApplication

What is the purpose, and how do the Configuration, ComponentScan, and Enable AutoConfiguration annotations work?

What is a profile in Spring Boot, and how can we use it?

How can we read values from the application.properties file?

Can we use Yaml and the properties file in one service? If yes, what will be the priority spring boot gives and for which file?

Have you used the @Qualifier annotation?

Have you used the spring boot rest annotation (@GetMapping, @PostMapping, @PatchMapping, @PutMapping, @DeleteMapping)?

how @PostMapping and @PatchMapping are different and their status codes.

What is @RestController?

Have you created a custom exception in Spring Boot? If yes, how?

how you are handling exceptions from the controller.

What are IOC, MVC, and DI?

How many ways can we perform dependency injection in spring?

What is an IOC container?

Have you worked in transaction management? Also, how many ways can we perform it?
Have you worked in security?
how you are handling security in your service.
What are authorization and authentication?
Have you used Spring Cloud?
How are you managing API Gateway? Please explain your project's microservice architecture.
What you are using for log purposes
what you are using for change in your service and how we can implement
what you are using for the unit test (JUnit, Mockito)
Are you maintaining code quality as well? (Yes, by using SonarQube and the Jacoco library.)

DB:

in which database you worked.
What is the difference between MongoDB and MySQL?
Why are you using MongoDB in your project?
What is an index in MongoDB?
Have you created a database pool or have you worked on transaction?
What is the aggregation function in MongoDB?
Have you worked with aggregation functions or have you used predefined functions from MongoDB, for example? \$gt, \$lt, and \$max
What is offset in MongoDB?
how you are writing queries in MongoDB.
What is join in MySQL?
How can we access two databases through a single service?

Microservice.

Can you explain microservice tools like GitLab, Jenkins, Docker, AWS, etc.?
Please let me know your project structure and module.
how you are calling one microservice to another microservice (RestTemplate, HttpClient, Feign client).
Have you ever had a problem connecting one microservice to another?
how you are handling API gateway, fault tolerance load balancing, service registry, and config client.

Kafka:

Are you using Kafka? If yes, have you worked in Kafka?
Why are you using Apache Kafka (for asynchronous calls) to reduce dependency between systems?
How does Apache Kafka work?
What is the means of gaining knowledge on Kafka topics?
How are you handling messages in Kafka if any exceptions or failures occur?
how you did configuration.
how you are consuming messages.
What are the Kafka key features (consumer, producer, topic, replica, partition, broker, zookeeper, offset, consumer group)?

What is the use of consumer groups?

how consumers will know which message was the last message and what they next need to read.

What is @kafkalistner?

Angular:

What is angular?

What are directives?

What was DI, and how did you handle it?

What is the decorator in angular?

How can we do two-way data binding?

What is the NG module? Can you explain the angular project architecture?

What are the pipes?

What is the component?

What is a service in Angular?