Commonly Asked Microservices Interview Questions

1>What do you understand by Microservices?

Microservices are an architectural style or approach used to build applications. The microservices architecture provides fast, frequent, and reliable deployment of large and complex applications. It is flexibly deployed and docked, so it won't break the entire application if you make any changes to a computer. Microservices are also known as microservices architecture, a variant of the structural style of Service Oriented Architecture (SOA) and used to structure an application as a collection of services that have the following characteristics:

Distributed and loosely coupled

Highly maintainable and verifiable

Independently deployable

Organized by business capabilities

2>What are the most significant benefits of using microservices?

The main benefit of using microservices is that you build an application to collect small self-contained services designed for an enterprise domain. So when the business constantly has to change, development teams can quickly create new application components to meet needs.

3>What are the main components of Microservices?

The following is the list of the main components of microservices or microservices architecture:

Containers, clustering and orchestration
IaC (Infrastructure as a concept of code)
Cloud infrastructure
API Gateway
Enterprise Service Bus
Service delivery

4>What are the main advantages of using Microservices?

Here is a list of some of the most important benefits of using microservices:

Microservices offer great technological diversity. You can easily mix it with other frameworks, libraries and databases.
microservices support error isolation by collecting small self-contained services or processes, so that one process failure does not have to bring down the entire system.
Provides excellent support for minor and parallel equipment.
Significantly reduces implementation times.
Independent distribution

5>Difference between monolithic and microservices?

1>Monolithic architecture is built as one large system and is usually one code-base

2>It is not easy to scale based on demand

3>It has shared database

4>Large code base makes IDE slow and build time gets increase.

5>It extremely difficult to change technology or language or framework because everything is tightly coupled and depend on each other

1>Microservices architecture is built as small independent module based on business functionality

2>It is easy to scale based on demand.

3>Each project and module has their own database

4>Each project is independent and small in size. So overall build and development time gets decrease.

5>Easy to change technology or framework because every module and project is independent

6>What is the use of containers in Microservices?

Containers are the easiest and most efficient way to manage microservice-based applications. They are like a software development platform. They also help us to develop and unfold ourselves individually. A docker is an example of a container. It is an open source software development platform that allows us to encapsulate our microservice along with its dependencies in a container image. The microservice can use these elements without additional effort.

7>How independent micro-services communicate with each other?

We can allow our microservices to communicate with each other according to the requirements of our project. For the most part, developers use HTTP / REST with JSON or binary protocol while using any communication protocol.
You can use either rest template or feign client.

8>What is Eureka in Microservices?

Eureka or Eureka Server is an application that contains the information about customer service applications. Microservices must be registered with the Eureka server, and the Eureka server knows all the client applications running on each port and each IP address. Alternatively, the Eureka server is also known as the Netflix service discovery server. It uses Spring Cloud and does not overload the application development process.

9>How can you balance the server-side load?

We can use Netflix Zuul to balance server-side load when using Spring Cloud. It is also known as a JVM-based router.

10> What is the use of Netflix Hystrix?

Hystrix is a latency and fault tolerance library. It is mainly used to isolate access points. It also ensures that all third-party libraries and services are limited. So the application works efficiently and avoids the kind of failures that occur in distributed systems.

11> What Is Semantic Monitoring?

Combine application-wide monitoring with automated testing. The main advantage of semantic tracking is discovering the factors that are most profitable for your business.

12> What is the use of PACT in Microservices architecture?

PACT is an open source tool used to test interactions between service providers and consumers. Increase the reliability of microservices applications.
Semantic monitoring along with service layer monitoring approximates microservices monitoring from a business perspective. Once a problem is identified, they allow for faster problem isolation and resolution, reducing primary time to repair. Evaluate the service and transaction layers to find out which transactions are affected by availability or poor performance.

13> What is a Client certificate?

A client certificate is a digital certificate that is used to send authenticated requests to a remote server. A certificate is generated for each microservice.

14> What is OAuth?

OAuth stands for Open Authorization Protocol. This protocol allows you to access client applications over HTTP for external providers GitHub, Facebook, etc. It also makes it easy for us to share resources stored on one site with another site without your credentials.

15> Bonus

Apart from these questions..
you should also prepare:
Springboot
spring cloud
Rest Api and
Spring Security interviеw question.
Interviewer may ask the question from these topic also.

Thanks for watching.