1. What is the difference between manual testing and automated testing?
In manual testing, testing of the software where performed manually by the QA analyst. The test is conducted to determine and discover bugs in the system. Analysts were tests the key and important
features of the software and prepares report about the tests which are performed and the results of the tests.

In automated software testing, testers write code / scripts to automate test execution. Testers use appropriate automation tools to develop the test scripts and validate the software. The main goal is to complete test execution in a lesser time. This kind of testing entirely relies on the pre scripted tests which runs automatically to compare actual results with the expected results.

2. What does assert class?
Assert class provides a set of assertion methods useful for writing tests.

3. How can be tested 'private' methods?
If you have somewhat of a legacy **Java** application, and you're not allowed to change the visibility of your methods, the best way to test private methods is to use reflection. Internally we're using helpers to get/set private and private static variables as well as invoke private and private static methods. The following patterns will let you do pretty much anything related to the private methods and fields. Of course, you can't change private static final variables through reflection.

4. **What is Monolithic Architecture?**
It' s a traditional unified model for the design of a software program. Which means composed all in one piece. It is designed to be self contained, components and the functions are tightly coupled. In a monolithic architecture, each component and its associated components must all be present for code to be executed or compiled for the software to run.

5. What are the best practices to write a Unit Test Case?

1. Write Readable Tests
2. Avoid magic numbers and magic strings
3. Write deterministic Tests
4. Avoid test interdepencies
5. Avoid logic in tests
6. Refrain multiple asserts in a single unit test
7. Keep your tests away from too much implementation details.
8. Write test during development, not after it.
9. Automate tests using CI/CD tools.
10. Update the tests periodically.

6. Why does JUnit only report the first failure in a single test?
Reporting multiple failures in a single test is generally a sign that the test does too much and it is too big a unit test. JUnit is designed to work best with a number of small tests.

7. What is the role of actuator in spring boot ?
Spring Boot's 'Actuator' dependency is used to monitor and manage the Spring

web application. We can use it to monitor and manage the application with the help of HTTP endpoints or with the JMX

8.What are the benefits and drawbacks of Microservices ?
Pros of Microservices:
1. Easier scaling up
2. Improved fault tolerance
3. Ease of understanding of the codebase
4. Scope for experimenting
5. Independent deployment

Cons of Microservices:
1. Increased complexity of communication
2. Requires more resources
3. Global testing and debugging is difficult
4. Not partiucalar for small applications
5. Relatively complex deployment.

9.What are the challenges that one has to face while using Microservices?
1.Design
2.Security
3.Testing
4.Operational Complexities
5.Communication

10.How independent microservices communicate with each other?
Because microservices are distributed and microservices communicate with each other by inter-service communication on network level. Each microservice has its own instance and process. Therefore, services must interact using an inter-service communication protocols like HTTP, gRPC or message brokers AMQP protocol.

11.What do you mean by Domain driven design ?
Domain-driven design (DDD) is a software development philosophy centered around the domain, or sphere of knowledge, of those that use it. The approach enables the development of software that is focused on the complex requirements of those that need it and doesn't waste effort on anything unneeded.

12. What is container in Microservices?
Containers are a form of operating system virtualization. A single container might be used to run anything from a small microservice or software process to a larger application. Inside a container are all the necessary executables, binary code, libraries, and configuration files.

13.What are the main components of Microservices architecture?
1. Microservices
2.Containers
3.Service mesh
4.Service discovery
5.API gateway

14.How does a Microservice architecture work?

The microservice architecture contains components depending on the business requirements. API Gateway- Clients need API Gateway as it is an entry point, which forwards the call to the specific services on the back end. Here API gateway helps in collecting the responses from different services and returns the response to the client. Microservices- As the name itself suggests that microservices are the services that help in dividing the service into small services that perform a certain business capability like user registration, current orders, or wish list. Database- Microservices can either share the same database or an independent database. Inter-microservices communication- REST or Messaging are the protocol to interact with each other. Now, let us learn more about the features of microservices.