

1-)What is the difference between manual testing and automated testing ?

In manual testing, a human performs the tests step by step, without test scripts. In automated testing, tests are executed automatically via test automation frameworks, along with other tools and software.

2-)What does Assert class?

The `assert()` function, which is essentially a macro, is used to generate diagnostics within the program. The resulting diagnostic information can be written to the standard error file. If the expression returns false, the `assert()` function displays an error message on `stderr` and stops the program from running.

3-)How can be tested 'private' methods ?

To test private methods, you just need to test the public methods that call them. Call your public method and make assertions about the result or the state of the object. If the tests pass, you know your private methods are working correctly.

4-)What is Monolithic Architecture ?

Monolithic architecture software is likely to be designed to be self-contained. We can also call it a standard “one piece”. Instead of components in this architecture, they are designed to be connected to each other.

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

1. Test One Thing at a Time in Isolation
2. Follow the AAA Rule: Arrange, Act, Assert
3. Write Simple “Fastball-Down-the-Middle” Tests First
4. Test Across Boundaries

5. If You Can, Test the Entire Spectrum
6. If Possible, Cover Every Code Path
7. Write Tests That Reveal a Bug, Then Fix It
8. Make Each Test Independent
9. Name Your Tests Clearly and Don't Be Afraid of Long Names
10. Test That Every Raised Exception Is Raised
11. Avoid the Use of Assert.IsTrue
12. Constantly Run Your Tests
13. Run Your Tests as Part of Every Automated Build

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. It executes each test within a separate instance of the test class.

7-)What are the benefits and drawbacks of Microservices?

Advantages of Microservices

- Microservices are self-contained, independent deployment module.
- The cost of scaling is comparatively less than the monolithic architecture.
- Microservices are independently manageable services. It can enable more and more services as the need arises. It minimizes the impact on existing service.
- It is possible to change or upgrade each service individually rather than upgrading in the entire application.
- Microservices allows us to develop an application which is organic (an application which latterly upgrades by adding more functions or modules) in nature.
- It enables event streaming technology to enable easy integration in comparison to heavyweight interposes communication.

- Microservices follows the single responsibility principle.
- The demanding service can be deployed on multiple servers to enhance performance.
- Less dependency and easy to test.
- Dynamic scaling.
- Faster release cycle.

Disadvantages of Microservices

- Microservices has all the associated complexities of the distributed system.
- There is a higher chance of failure during communication between different services.
- Difficult to manage a large number of services.
- The developer needs to solve the problem, such as network latency and load balancing.
- Complex testing over a distributed environment.

8-)What is the role of actuator in spring boot ?

Spring boot's actuator module allows us to monitor and manage application usages in production environment, without coding and configuration for any of them. These monitoring and management information is exposed via REST like endpoint URLs. ... Returns a complete list of all the Spring beans in your application.

9-)What are the challenges that one has to face while using Microservices ?

Microservices architecture is great architectural patterns but it comes with some of the common challenges, as the number of services grows so does the complexity of managing services and maintaining data consistency. Testing also becomes a bottleneck due to dependency on other services.

Microservices create different types of complexity than monolithic applications for development teams. First, communication between services can be complex. Interface control is even more critical. Up-front costs may be higher with microservices.

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 is the idea of solving problems of the organization through code. The business goal is important to the business users, with a clear interface and functions. This way, the microservice can run independently from other microservices.

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 ?

- Clients.
- Identity Providers.
- API Gateway.
- Messaging Formats.
- Databases.
- Static Content.
- Management.
- Service Discovery.

14-)How does a Microservice architecture work?

Microservice architecture, aka microservices, are a specific method of designing software systems to structure a single application as a collection of loosely coupled services. Applications tend to begin as a monolithic architecture (more on that below), and over time grow into a set of interconnected microservices.