What is a microservice?

* A design used primarily in functional programming and object-oriented programming
* A small program that represents discrete logic that executes within a well-defined boundary on dedicated hardware
* A style of design for enterprise systems based on a loosely coupled component architecture

**More about the answer:**

In its simplest form, [a microservices](https://searchapparchitecture.techtarget.com/definition/microservices) architecture consists of smaller, loosely-coupled components that combine to form a larger application.

* A very small piece of code that never gets any bigger than 10 lines

**Question 2 of 10**

When would developers use microservices?

* When they want to write cell phone applications that run quickly
* When they work with ephemeral nano technology
* When they need to create large, enterprise-level applications that are subject to changes on a frequent basis

**More about the answer:**

Microservices allow enterprises to [scale quickly and rapidly](https://searchapparchitecture.techtarget.com/tip/Is-microservices-adoption-right-for-your-organization) when properly adopted.

* When they create applications specifically for scientific test equipment

**Question 3 of 10**

Which of the following responses is an advantage of microservices?

* Any microservice component can change independently from other components

**More about the answer:**

The loosely coupled architecture allows a developer to change one element of a [microservices architecture design](https://searchapparchitecture.techtarget.com/tip/Follow-these-10-fundamental-microservices-design-principles) -- a data log, for example -- without affecting other elements.

* They don't require a lot of expertise to program
* They're so small that developers can typically write very powerful ones with a few lines of text
* They are easy to manage

**Question 4 of 10**

Which of the following responses is a disadvantage of microservices?

* Microservices are very difficult to manage at scale
* Microservices require a lot of monitoring to operate effectively
* Neither A nor B
* Both A and B

**More about the answer:**

A microservices architecture won't be easy. [These challenges](https://searchcloudcomputing.techtarget.com/tip/Overcome-9-microservices-challenges-for-platform-flexibility) require dedicated personnel and a high overhead to run smoothly.

**Question 5 of 10**

What are some technologies commonly used to implement microservices?

* Docker
* Kubernetes
* All the above

**More about the answer:**

Containers, service mesh and API gateways make up some of the [most common components](https://searchapparchitecture.techtarget.com/feature/5-core-components-of-microservices-architecture) of microservices. All these technologies are commonly found in a microservices architecture.

* None of the above

**Question 6 of 10**

What is a popular Java framework to develop microservices?

* Spring Boot
* Eclipse MicroProfile
* Both A and B

**More about the answer:**

Spring Boot and Eclipse MicroProfile are two of the many Java frameworks to developer microservices, but they [aren't the only options](https://searchapparchitecture.techtarget.com/feature/6-quick-facts-about-a-Java-microservices-architecture).

* None of the above

**Question 7 of 10**

How is distributed tracing used in microservices?

* As a mechanism to ensure that failed microservices are resurrected properly
* As a mechanism to transfer log management between a given host on demand
* As a mechanism to observe the behavior of distinct system calls between and within microservices

**More about the answer:**

[Distributed tracing tools](https://searchapparchitecture.techtarget.com/tip/3-distributed-tracing-tools-perfect-for-microservices) help developers map and isolate data across system pathways to trace requests and help identify potential bottlenecks.

* As a mechanism to change the behavior of a microservice at runtime

**Question 8 of 10**

How does a backing service apply to microservices?

* It prevents a microservice from failing
* It acts as a dedicated service that provides essential functionality required by a microservice

**More about the answer:**

The backing service works as a fail-safe option to [maintain functional elements](https://searchapparchitecture.techtarget.com/tip/3-lessons-microservices-developers-can-learn-from-SOA) of a microservices component and allows the developer to continue building a resilient application.

* It shuts down a microservice when it can no longer handle the computing load
* It coordinates network activity between microservices

**Question 9 of 10**

What's the difference between a microservices-oriented architecture (MOA) and a service-oriented architecture (SOA)?

* An SOA uses intermediation technology to facilitate communication between services
* An MOA shares as little data as possible while an SOA shares as much data as possible
* A developer can run a monolithic application with SOA principles
* All the above

**More about the answer:**

[Implementation is the main difference](https://searchapparchitecture.techtarget.com/tip/5-basic-SOA-principles-that-still-apply-to-microservices?_gl=1*ly6nsh*_ga*MTU2MDY0OTA2LjE2NTc3NzMwNzY.*_ga_TQKE4GS5P9*MTY2MjMwMjUwMS43LjEuMTY2MjMwMjU5Ni4wLjAuMA..&_ga=2.245307776.489612973.1662302501-156064906.1657773076) between MOA and SOA. While both share compartmentalized design principles, how developers choose to interconnect components, self-sufficiency, technologies and data sharing differ significantly between the two.

**Question 10 of 10**

What does the term "bounded context" mean in relation to microservices?

* How a microservice uses memory
* The logical domain represented by the data consumed and emitted by a microservice according to the data's purpose, structure and meaning

**More about the answer:**

Bounded contexts prevent a microservice component from [overstepping its bounds](https://docs.microsoft.com/en-us/azure/architecture/microservices/model/domain-analysis). While domains are connected, a microservices component should only do its work within its domain and share that data after completion.

* The cloud provider's region where the microservice operates
* The common characteristics within a collection of entities in one or many business domains