

1. Which phase of the Twelve-Factor Methodology contains Port binding?

- ☒ Deploy
- ☐ Configure
- ☐ Operate
- ☐ Code

✓ **Correct**

Correct. Deploy phase of the delivery lifecycle contains: Config, Backing services, Processes, and Port binding.

2. What makes an application concurrent?

- ☐ It should store config in environment variables
- ☐ It should have a minimal process start time
- ☐ It should contain the same backend services across environments
- ☒ It should be stateless

✓ **Correct**

Correct. If processes are stateless and share nothing, an application can start additional processes to scale horizontally and handle the additional workload without creating interdependencies among processes.

3. In a microservice architecture, you should

- ☒ Use a different technology stack for each service
- ☐ Scale all services together
- ☐ Deploy all services together
- ☐ Create interdependency among services

✓ **Correct**

Correct. Teams can choose even different programming languages for different components as they are dependent on each other via an API endpoint.

4. Select the correct answer that completes this sentence:

Monolith application has _____ components.

- ☐ Scalable
- ☐ Independent
- ☐ Reusable
- ☒ Interconnected

✓ **Correct**

Correct. A monolithic application has all or most of its functionality within a single process. Making them tightly connected and dependent on each other.

5. Which component of services design in an SOA architecture defines how the service provider and service consumer should interact?

- ☐ Interface
- ☒ Contract
- ☐ Implementation
- ☐ Own storage

✓ **Correct**

Correct. A contract defines how the service provider and service consumer should interact.

6. What is the purpose of Backend For Frontend?

- ☒ Customized user experiences
- ☐ Loading one interface that never reloads
- ☐ Service discovery
- ☐ Refactoring in stages

✓ **Correct**

Correct. The BFF pattern allows developers to create and support one backend type per user interface using the best options for that interface rather than trying to support a generic backend that works with any interface but may negatively impact frontend performance.

7. What is the purpose of the Strangler pattern?

- ☐ Simplifies the front-end experience
- ☒ Supports refactoring in stages
- ☐ Helps with third-party API integrations
- ☐ Combine functional domains

✓ **Correct**

Correct. With the Strangler pattern, you use the structure of a web application to split an application into multiple functional domains and replace those domains with a new microservices-based implementation for one domain at a time.

8. Larger services should be broken into smaller services when:

- ☐ Automation is required
- ☐ Functional reusability is required
- ☐ Refactoring is required
- ☒ Common data model becomes overly complex

✓ **Correct**

Correct. If your data model becomes too broad, it is time to split the service and functionally segregate them.

9. What is the requirement of Microservices?

☐ Independent storage

10. What is the requirement of a development and production environment?

- ☐ Different source control
- ☐ Different dependencies
- ☐ Embedded access details in code
- ☒ Similar backing services across the environments.

✓ **Correct**

Correct. If you use a MySQL database in production, you should use the same version of MySQL database in your development environments.