Download Linux Foundation CGOA Exam Dumps Free

Shared by Cote on 17-06-2026

For More Free Questions and Preparation Resources

Check the Links on Last Page

Question 1

Question Type: MultipleChoice

Which requirement of the GitOps principle declares that Desired State must be versioned?

Options:

A- The Desired State must not be publicly accessible.
B- The State Store must retain a complete version history.
C- You must use Git in order to be compliant with this principle.
D- The Desired State must be publicly accessible.

Answer:

B

Explanation:

One of the GitOps principles is Versioned and Immutable, which requires that the Desired State is stored in a system that maintains a complete version history. This allows for auditing, traceability, and rollback.

''The Desired State must be stored in a versioned, immutable system. The State Store must retain a complete version history so changes can be audited and previous states can be restored.''

Thus, the correct answer is B.

==========

Question 2

Question Type: MultipleChoice

When using Kustomize, how are resources, configurations, and customizations commonly organized?

Options:

A- By specifying all resources inline in the customization file.
B- In separate configuration files for each resource.
C- In a single configuration file.
D- Using a combination of folder directories and referenced folder/file paths.

Answer:

D

Explanation:

Kustomize is a GitOps tool for managing Kubernetes configurations declaratively. It uses a folder structure with configuration files and a kustomization.yaml file that references resources and overlays. This enables customization without modifying the base manifests.

''Kustomize allows customization of Kubernetes manifests by organizing resources in directories and referencing them through file paths in a kustomization file. This directory-based approach supports overlays, reusability, and modular configuration.''

Thus, the correct answer is D.

==========

Question 3

Question Type: MultipleChoice

In GitOps, what does it mean to Continuously Reconcile?

Options:

A- Regularly update Git repositories with the latest changes from external sources.
B- Perform regular backups of Git repositories.
C- Automatically compare and adjust the system state as needed.
D- Monitor the system for any unauthorized changes and revert them.

Answer:

C

Explanation:

Continuous reconciliation is another core GitOps principle. It means that software agents (operators or controllers) run loops that continuously observe the live system and compare it against the desired state declared in Git. If any divergence (drift) is found, the agent automatically reconciles the system to match the declared configuration.

''Software agents continuously observe the actual system state and compare it with the desired state declared in Git. If a divergence is detected, the agents automatically reconcile the difference to bring the system back into alignment.''

This provides automation, consistency, and self-healing, which are hallmarks of GitOps.

Question 4

Question Type: MultipleChoice

In GitOps, how is the Desired State stored?

Options:

A- In a way that enforces mutability and versioning.
B- In a way that permits direct modifications to live systems.
C- In a way that retains only the latest version.
D- In a way that enforces immutability and versioning.

Answer:

D

Explanation:

The GitOps principle of Versioned and Immutable requires Desired State to be stored in a way that enforces immutability and versioning. This ensures every change is recorded, auditable, and reversible.

''Desired state must be kept in an immutable, version-controlled system. This guarantees a full history of changes and enables safe rollbacks.''

Thus, the correct answer is D.

==========

Question 5

Question Type: MultipleChoice

What is one of the key benefits of a pull-based reconciliation approach to configuration management?

Options:

A- Simplified troubleshooting and debugging processes.
B- Agents can access the Desired State at any time, not only when an event is triggered.
C- The CI has access credentials to the running system.
D- Immediate response time to configuration changes.

Answer:

B

Explanation:

In GitOps, the pull-based reconciliation approach means that agents continuously monitor the Desired State in Git. Unlike push-based systems, which only act when triggered, pull-based systems can reconcile at any time, providing resilience, self-healing, and security (since no external system needs direct access to the cluster).

''In a pull-based model, reconciliation agents continuously fetch and compare the desired state, enabling self-healing and ensuring the desired configuration is accessible at all times.''

Thus, the correct answer is B.

Question 6

Question Type: MultipleChoice

How does GitOps handle drift during reconciliation?

Options:

A- Write Kubernetes Patch files in a database for later use.
B- Attempt to apply Desired State to the running system.
C- Write back to Desired State to match the actual state.
D- Find the differences between Desired State and actual state and create a new system based on these changes.

Answer:

B

Explanation:

When drift occurs (actual state diverges from desired state), GitOps controllers attempt to reapply the Desired State stored in Git. The system is always converged toward what Git declares, never the other way around.

''In case of drift, the reconciler re-applies the desired state from Git to the runtime environment, ensuring the actual system matches the declared configuration.''

Thus, the correct answer is B.

Question 7

Question Type: MultipleChoice

You want to create a dashboard to monitor the performance of your application. Which of the following is a key principle of GitOps regarding dashboards?

Options:

A- The operations team should manually update dashboards.
B- Dashboards should be created using a proprietary tool.
C- Dashboards should only be accessible to the development team.
D- Dashboards declarations should be in the Desired State store.

Answer:

D

Explanation:

In GitOps, everything that defines the system, including dashboards, must be stored declaratively in Git (the Desired State store). This ensures dashboards are versioned, reproducible, and consistent across environments.

''GitOps requires that all system components, including monitoring and observability configurations such as dashboards, are declared in Git. This ensures they are versioned, immutable, and reproducible.''

Thus, D is correct.

==========

Question 8

Question Type: MultipleChoice

Which of these is an advantage of using a declarative configuration for your Desired State?

Options:

A- Declarative configuration allows you to execute code locally more efficiently to make desired changes to your running system.
B- Using widely adopted community tools for reconciling actual state is less work than maintaining custom imperative scripts.
C- Declarative configuration helps you include dynamic scripting that guides an application through a step-by-step process.
D- Declarative configuration lets you specify complex if/else logic within custom code.

Answer:

B

Explanation:

Declarative configuration describes what the system should look like, not how to achieve it. This enables the use of standard reconciliation tools (like ArgoCD or Flux) to manage the system automatically, removing the burden of writing and maintaining imperative scripts.

''Declarative configuration enables systems to be managed by generic reconciliation tools rather than bespoke scripts, reducing operational overhead and increasing reliability.''

Thus, the correct answer is B.

==========

Question 9

Question Type: MultipleChoice

In GitOps practices, when does CD take part?

Options:

A- CD takes part simultaneously with CI, both components of GitOps practices.
B- CD takes part after CI to automate the deployment of applications based on changes in the Git repository.
C- CD takes part before CI stage in order to ensure the successful deployment of applications.
D- CI plays a significant role in GitOps practices.

Answer:

B

Explanation:

In GitOps, Continuous Deployment (CD) follows after Continuous Integration (CI). CI is responsible for building and testing application code, while CD automates the delivery and deployment of these changes into runtime environments. The Git repository serves as the single source of truth, and when CI merges new changes into the main branch, CD reconciles the state of the environment to match what is declared in Git.

''GitOps builds on the principles of DevOps by using Git as the source of truth for declarative infrastructure and applications. CI pipelines handle the integration and testing of code, and CD pipelines or agents automatically reconcile the desired state in Git with the actual state in the cluster.''

This shows that CD is triggered after CI to handle deployment automation, ensuring systems remain in sync with what is declared in version control.

Question 10

Question Type: MultipleChoice

What is the main difference between Terraform/OpenTofu and Ansible?

Options:

A- Terraform/OpenTofu uses a configuration language called CUE, while Ansible uses HCL.
B- Terraform/OpenTofu stores the state of each resource, while Ansible works in a fire-and-forget mode.
C- Terraform/OpenTofu is imperative in nature, while Ansible is declarative.
D- Ansible is written in Golang, while Terraform/OpenTofu is written in Python.

Answer:

B

Explanation:

Terraform (or OpenTofu) uses a declarative model and maintains a state file to track the current status of resources, enabling it to plan and reconcile changes. Ansible, by contrast, is more procedural and executes tasks in a fire-and-forget manner, without tracking persistent resource state.

''Terraform maintains state for each managed resource, enabling planned, consistent changes. Ansible executes tasks without tracking resource state, working in a fire-and-forget model.''

Thus, the correct answer is B.

==========

Question 11

Question Type: MultipleChoice

Given GitOps, what does Desired State refer to?

Options:

A- The state that the system or application should be in.
B- The state that the system or application was in before any changes were made.
C- The current state of the system or application.
D- The state that the system or application will be in after all changes are made.

Answer:

A

Explanation:

The Desired State is the declarative specification stored in Git that defines how the system should look and behave. It is the reference point against which the actual state is continuously reconciled.

''Desired state is the complete declarative specification of a system stored in Git. It defines how the system should be configured and serves as the source of truth for reconciliation.''

Thus, the correct answer is A.