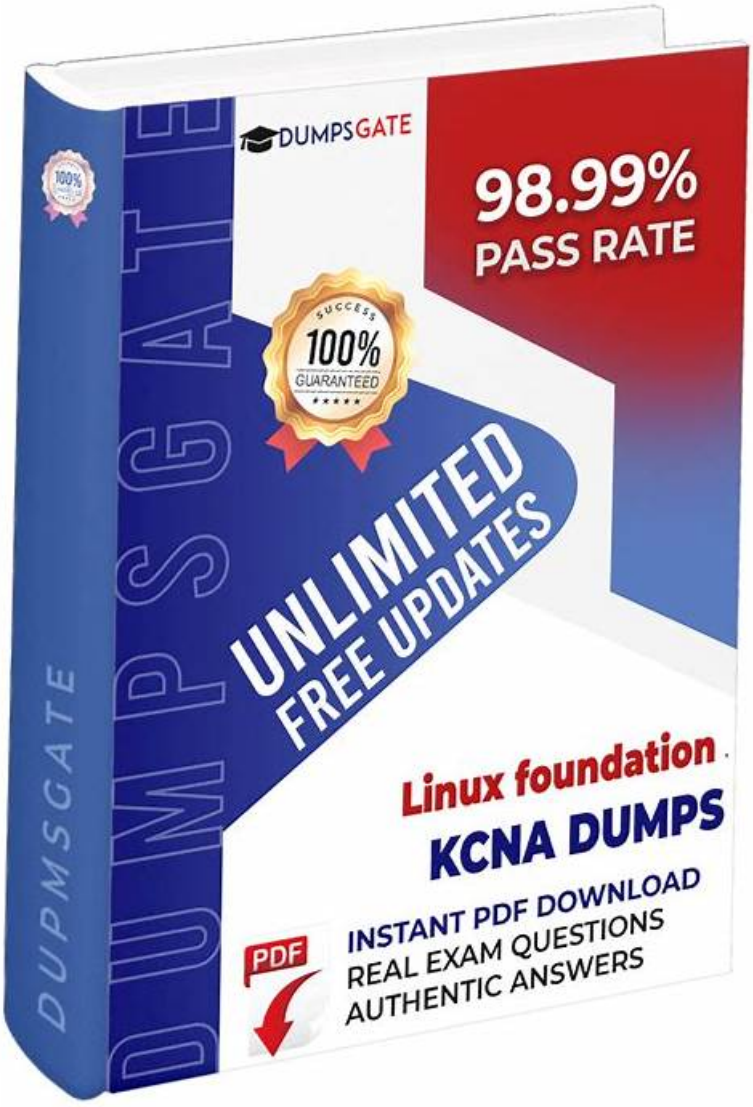
Cost Effective KCNA Dumps | New KCNA Test Experience



DOWNLOAD the newest ITexamReview KCNA PDF dumps from Cloud Storage for free: https://drive.google.com/open?id=118QO9IL8P6dKPJxyLJliUUsuN0ULpcTg

We update the KCNA study materials frequently to let the client practice more and follow the change of development in the practice and theory. So that our worthy customers can always receive the most updated and the latest KCNA learning guide. And according to our service, you can enjoy free updates for one year after you pay for the KCNA Exam Questions. So if we update it, then we will auto send it to you. You won't miss any information that you need to pass the exam.

Linux Foundation recommends that individuals who are interested in taking the KCNA certification exam have some prior knowledge of Linux and basic cloud computing concepts. However, it is not a requirement, and anyone can take the exam regardless of their background. KCNA Exam is designed to be accessible to everyone, and the Linux Foundation provides resources and training materials to help individuals prepare for the exam.

>> Cost Effective KCNA Dumps <<

2026 Reliable Cost Effective KCNA Dumps | Kubernetes and Cloud Native Associate 100% Free New Test Experience

All our regular candidates have impulse to choose again when they have the similar KCNA exam. So they totally trust us. All exams are not insuperable obstacle anymore with our KCNA training materials. Our credibility is unquestionable. In the course of obtaining success, we need a number of helps, either external or internal, but to the exam, the quality of KCNA practice materials are of great importance. So our KCNA learning dumps are acclaimed as masterpieces.

Linux Foundation Kubernetes and Cloud Native Associate Sample Questions (Q138-Q143):

NEW QUESTION # 138
What is the main difference between Argo vs. Flux CD?

- A. No difference; both are pull-based
- B. No difference; both are push-based
- C. Argo is push-based, and Flux is pull-based
- D. Argo is pull-based, and Flux is push-based

Answer: C

Explanation:
AgroCD: https://argo-cd.readthedocs.io/en/stable/developer-guide/ci/#can-i-retrigger-the-checks-without-pushing-a-new-commit
FluxCD: https://fluxcd.io/

NEW QUESTION # 139
Your application requires a specific storage class for its persistent dat
a. How do you configure this storage class within your deployment YAML?

- A. Specify the storage class name within the field of the deployment.
- B. Specify the storage class name directly within the 'spec.template.spec.containers[0].volumeMounts[0].name' field of the deployment.
- C. Create a separate PersistentVolumeClaim (PVC) with the desired storage class and reference the PVC in the deployment's name' field.
- D. None of the above
- E. Specify the storage class name within the 'spec.template.spec.containers[01.volumeMounts[0].storageClassName' field of the deployment.

Answer: C

Explanation:
The correct approach is to create a separate PersistentVolumeClaim (PVC) that specifies the desired storage class and reference the PVC in the deployments 'spec_template.spec_containers[0]_volumeMounts[0]_name' field. This ensures the PVC is automatically bound to a PV with the correct storage class. Specifying the storage class name directly within the deployment or the volumeMounts section is not the standard practice for defining storage requirements.

NEW QUESTION # 140
Your application relies on a backend database service. Using Istio, you want to configure a circuit breaker pattern to prevent cascading failures if the database becomes unresponsive. How would you implement this?

- A. Deploy a separate health check pod to monitor the database service and trigger the circuit breaker if needed
- B. Utilize Istio's 'fault injection' feature to simulate failures and test the circuit breaker
- C. Create a custom Istio VirtualService with a 'destinationRule' specifying a fallback service in case of failure
- D. Configure the 'retry' policy in the Istio configuration to automatically retry failed requests
- E. Use the 'timeout' setting in the Istio configuration to limit the duration of requests to the database

Answer: C

Explanation:
Istio's VirtualService and DestinationRule features allow you to configure fallback services. In the case of the database being unavailable, you can define a fallback service or mechanism to handle the request. Option B helps test circuit breaker behavior but doesn't implement it. Option C could be used for retries, but doesn't address the circuit breaker pattern. Option D is related to

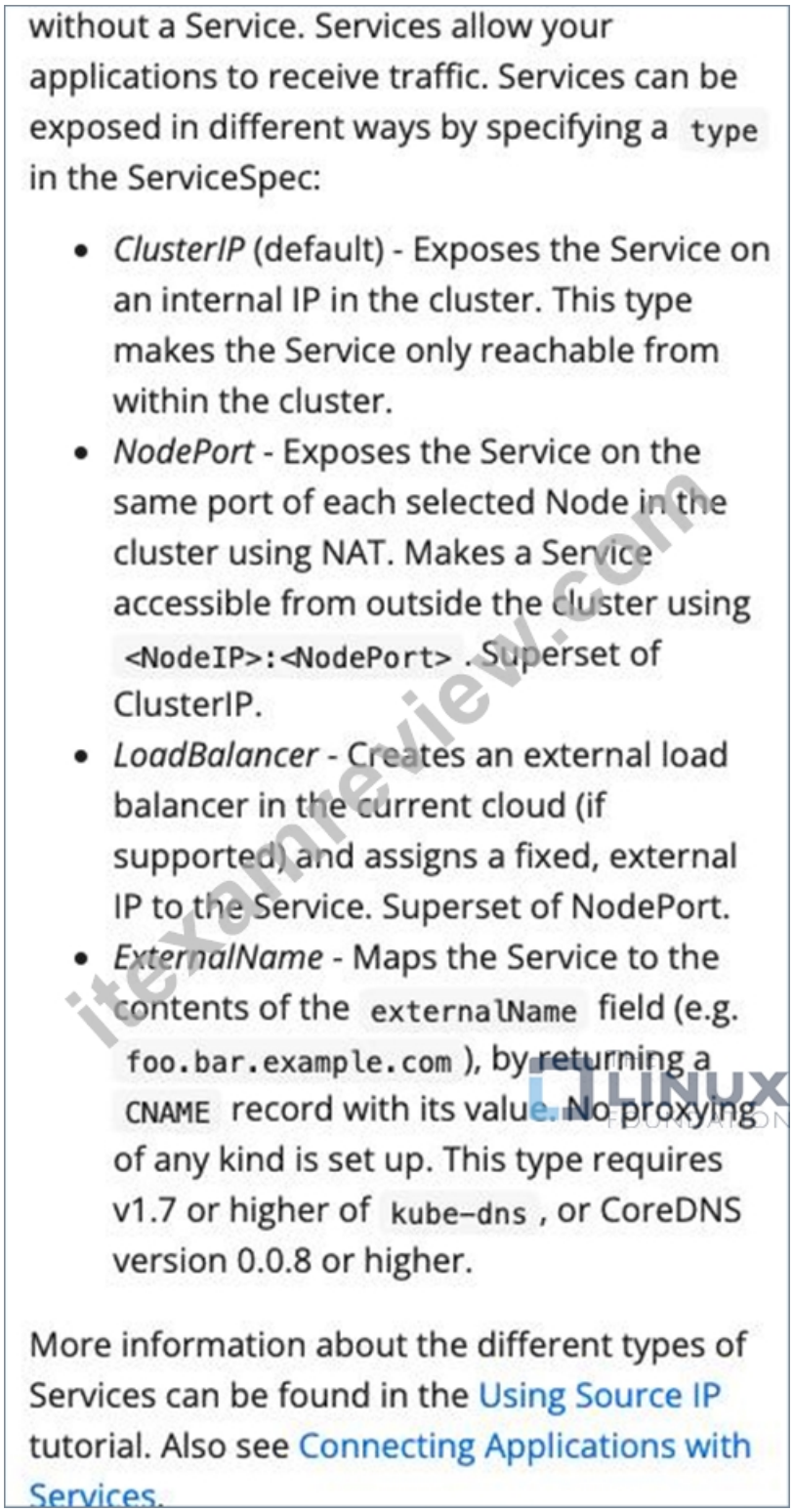
request timeouts, not circuit breaking. Option E is a manual approach, while Istio provides a more integrated solution.

NEW QUESTION # 141
Which is not a service type in Kubernetes?

- A. Ingress
- B. LoadBalancer
- C. ClusterIP
- D. NodePort
- E. ExternalName

Answer: A

Explanation:
https://kubernetes.io/docs/tutorials/kubernetes-basics/expose/expose-intro/



NEW QUESTION # 142
In CNCF, who develops specifications for industry standards around container formats and runtimes?

- A. Open Container Initiative (OCI)
- B. Container Runtime Interface (CRI)
- C. Linux Foundation Certification Group (LFCG)
- D. Container Network Interface (CNI)

Answer: A

Explanation:

The organization responsible for defining widely adopted standards around container formats and runtime specifications is the Open Container Initiative (OCI), so A is correct. OCI defines the image specification (how container images are structured and stored) and the runtime specification (how to run a container), enabling interoperability across tooling and vendors. This is foundational to the cloud-native ecosystem because it allows different build tools, registries, runtimes, and orchestration platforms to work together reliably.
Within Kubernetes and CNCF-adjacent ecosystems, OCI standards are the reason an image built by one tool can be pushed to a registry and pulled/run by many different runtimes. For example, a Kubernetes node running containerd or CRI-O can run OCI-compliant images consistently. OCI standardization reduces fragmentation and vendor lock-in, which is a core motivation in open source cloud-native architecture.
The other options are not correct for this question. CNI (Container Network Interface) is a standard for configuring container networking, not container image formats and runtimes. CRI (Container Runtime Interface) is a Kubernetes-specific interface between kubelet and container runtimes-it enables pluggable runtimes for Kubernetes, but it is not the industry standard body for container format/runtime specifications. "LFCG" is not the recognized standards body here.
In short: OCI defines the "language" for container images and runtime behavior, which is why the same image can be executed across environments. Kubernetes relies on those standards indirectly through runtimes and tooling, but the specification work is owned by OCI. Therefore, the verified correct answer is A.

NEW QUESTION # 143
.....

myportal.utt.edu.tt, Disposable vapes

What's more, part of that ITexamReview KCNA dumps now are free: https://drive.google.com/open?id=118QO9IL8P6dKPJxyLJliUUsuN0ULpcTg