

New Linux Foundation KCNA Braindumps Pdf, KCNA Questions Exam

NEWEXAMDUMPS

Linux Foundation

KCNA
Kubernetes and Cloud Native Associate
QUESTION & ANSWERS

P.S. Free & New KCNA dumps are available on Google Drive shared by Pass4training: <https://drive.google.com/open?id=1JQmQhvmZESHMh2DiflGIT5cFuWsrgWGm>

By concluding quintessential points into Kubernetes and Cloud Native Associate practice materials, you can pass the exam with the least time while huge progress. Our experts are responsible to make in-depth research on the exams who contribute to growth of our KCNA practice materials. Their highly accurate exam point can help you detect flaws on the review process and trigger your enthusiasm about the exam. What is more, KCNA practice materials can fuel your speed and the professional backup can relieve you of stress of the challenge.

Our KCNA exam torrent is available in different versions. Whether you like to study on a computer or enjoy reading paper materials, our test prep can meet your needs. Our PDF version of the KCNA quiz guide is available for customers to print. You can print it out, so you can practice it repeatedly conveniently. And our KCNA exam torrent make it easy for you to take notes on it so that your free time can be well utilized and you can often consolidate your knowledge. Everything you do will help you successfully pass the exam and get the card. The version of APP and PC of our KCNA Exam Torrent is also popular. They can simulate real operation of test environment and users can test KCNA test prep in mock exam in limited time. They are very practical and they have online error correction and other functions. The characteristic that three versions of KCNA exam torrent all have is that they have no limit of the number of users, so you don't encounter failures anytime you want to learn our KCNA quiz guide. The three different versions can help customers solve any questions and meet their all needs.

>> **New Linux Foundation KCNA Braindumps Pdf** <<

Linux Foundation KCNA Dumps PDF Format

In order to meet the need of all customers, there are a lot of professionals in our company. We can promise that we are going to provide you with 24-hours online efficient service after you buy our Kubernetes and Cloud Native Associate guide torrent. We are willing to help you solve your all problem. If you purchase our KCNA test guide, you will have the right to ask us any question about our products, and we are going to answer your question immediately, because we hope that we can help you solve your problem about our KCNA Exam Questions in the shortest time. We can promise that our online workers will be online every day. If you buy our KCNA test guide, we can make sure that we will offer you help in the process of using our KCNA exam questions. You will have the opportunity to enjoy the best service from our company.

Linux Foundation Kubernetes and Cloud Native Associate Sample Questions (Q131-Q136):

NEW QUESTION # 131

You have a Kubernetes deployment that runs an application with multiple replicas. When a Pod fails, you need to ensure that the deployment automatically creates a replacement Pod. Which Kubernetes API resource is responsible for this functionality?

- A. Deployment
- **B. ReplicaSet**
- C. StatefulSet
- D. pod
- E. Service

Answer: B

Explanation:

The ***ReplicaSet*** is responsible for ensuring that the desired number of Pods are running for a deployment. It monitors the Pods and automatically replaces failed Pods. Option 'A' is incorrect because services are for exposing applications, not for managing replicas. Option 'C' is incorrect because Pods are individual containers. Option 'D' is incorrect because Deployments use ReplicaSets to manage their replicas. Option 'E' is incorrect because StatefulSets are for managing Pods with unique identities and persistent storage, which is not required in this scenario.

NEW QUESTION # 132

The Kubernetes rolling update is used for ___.

- **A. Updating a deployment**
- B. Scaling an application
- C. Updating a service

Answer: A

Explanation:

<https://kubernetes.io/docs/tutorials/kubernetes-basics/update/update-intro/>

NEW QUESTION # 133

You are managing a Kubernetes cluster with several nodes. You notice that one node is experiencing high CPU utilization due to a specific pod. How can you force this pod to be moved to a different node without deleting or restarting it?

- A. Use the 'kubectl cordon' command to prevent new pods from being scheduled on the overloaded node.
- **B. Use the 'kubectl drain' command to remove all pods from the node, allowing you to reschedule the specific pod.**
- C. Use the 'kubectl delete pod' command to delete the pod, causing it to reschedule on another node.
- D. Manually edit the pod's YAML configuration and change the 'nodeName' field to a different node in the cluster
- E. Use the 'kubectl exec -it - bash' command to access the pod's shell and manually migrate the container to another node.

Answer: B

Explanation:

The 'kubectl drain' command is used to gracefully remove all pods from a node, preparing it for maintenance or eviction. The command will not delete the pods, allowing them to be rescheduled on other nodes. This allows you to move the pod without deleting or restarting it. Option A deletes the pod, which is not what the question asks for. Options B and C are not possible or recommended actions for pod migration. Option D prevents new pods from scheduling on the overloaded node but won't force the

pod to move.

NEW QUESTION # 134

What is a Service?

- A. A static network mapping from a Pod to a port.
- B. An NGINX load balancer that gets deployed for an application.
- C. The network configuration for a group of Pods.
- **D. A way to expose an application running on a set of Pods.**

Answer: D

Explanation:

The correct answer is B; a Kubernetes Service is a stable way to expose an application running on a set of Pods. Pods are ephemeral-IPs can change when Pods are recreated, rescheduled, or scaled. A Service provides a consistent network identity (DNS name and usually a ClusterIP virtual IP) and a policy for routing traffic to the current healthy backends.

Typically, a Service uses a label selector to determine which Pods are part of the backend set. Kubernetes then maintains the corresponding endpoint data (Endpoints/EndpointSlice), and the cluster dataplane (kube-proxy or an eBPF-based implementation) forwards traffic from the Service IP/port to one of the Pod IPs. This enables reliable service discovery and load distribution across replicas, especially during rolling updates where Pods are constantly replaced.

Option A is incorrect because Service routing is not a "static mapping from a Pod to a port." It's dynamic and targets a set of Pods.

Option C is too vague and misstates the concept; while Services relate to networking, they are not "the network configuration for a group of Pods" (that's closer to NetworkPolicy/CNI configuration). Option D is incorrect because Kubernetes does not automatically deploy an NGINX load balancer when you create a Service. NGINX might be used as an Ingress controller or external load balancer in some setups, but a Service is a Kubernetes API abstraction, not a specific NGINX component.

Services come in several types (ClusterIP, NodePort, LoadBalancer, ExternalName), but the core definition remains the same: stable access to a dynamic set of Pods. This is foundational for microservices and for decoupling clients from the churn of Pod lifecycles.

So, the verified correct definition is B.

NEW QUESTION # 135

Which of the following is a feature Kubernetes provides by default as a container orchestration tool?

- A. File system redundancy.
- B. A container image registry.
- **C. Automated rollouts and rollbacks.**
- D. A portable operating system.

Answer: C

Explanation:

Kubernetes provides automated rollouts and rollbacks for workloads by default (via controllers like Deployments), so D is correct. In Kubernetes, application delivery is controller-driven: you declare the desired state (new image, new config), and controllers reconcile the cluster toward that state. Deployments implement rolling updates, gradually replacing old Pods with new ones while respecting availability constraints. Kubernetes tracks rollout history and supports rollback to previous ReplicaSets when an update fails or is deemed unhealthy.

This is a core orchestration capability: it reduces manual intervention and makes change safer. Rollouts use readiness checks and update strategies to avoid taking the service down, and kubectl rollout status/history/undo supports day-to-day release operations. The other options are not "default Kubernetes orchestration features":

Kubernetes is not a portable operating system (A). It's a platform for orchestrating containers on top of an OS.

Kubernetes does not provide filesystem redundancy by itself (B). Storage redundancy is handled by underlying storage systems and CSI drivers (e.g., replicated block storage, distributed filesystems).

Kubernetes does not include a built-in container image registry (C). You use external registries (Docker Hub, ECR, GCR, Harbor, etc.). Kubernetes pulls images but does not host them as a core feature.

So the correct "provided by default" orchestration feature in this list is the ability to safely manage application updates via automated rollouts and rollbacks.

