KCNA - Sample Practice Questions

KCNA Sample Practice Questions

Download PDF Version

> **Disclaimer**: These are sample practice questions created for study purposes only. They are NOT actual exam questions and are designed to help you test your understanding of KCNA concepts. Real exam questions may differ in format and content.

Instructions

- Each question has one correct answer unless otherwise specified
- Try to answer without looking at the solutions first
- Review the explanations to understand the concepts better

Section 1: Kubernetes Fundamentals (46%)

Question 1.1

What is the smallest deployable unit in Kubernetes?

A) Container
B) Pod
C) Deployment
D) ReplicaSet

Show Answer
Answer: B) Pod A Pod is the smallest deployable unit in Kubernetes. It can contain one or more containers that share storage and network resources.

Question 1.2

Which control plane component is responsible for storing all cluster data?

A) kube-apiserver
B) kube-scheduler
C) etcd
D) kube-controller-manager

Show Answer
Answer: C) etcd etcd is a consistent and highly-available key-value store used as Kubernetes' backing store for all cluster data.

Question 1.3

What is the default Service type in Kubernetes?

A) NodePort
B) LoadBalancer

C) ClusterIP
D) ExternalName

Show Answer
Answer: C) ClusterIP ClusterIP is the default Service type. It exposes the Service on a cluster-internal IP, making it only reachable from within the cluster.

Question 1.4

Which component runs on every node and ensures containers are running in a Pod?

A) kube-proxy
B) kubelet
C) kube-scheduler
D) container runtime

Show Answer
Answer: B) kubelet The kubelet is an agent that runs on each node in the cluster. It ensures that containers are running in a Pod as specified.

Question 1.5

What Kubernetes object would you use to store non-confidential configuration data?

A) Secret
B) ConfigMap
C) PersistentVolume
D) ServiceAccount

Show Answer
Answer: B) ConfigMap ConfigMaps are used to store non-confidential data in key-value pairs. Secrets are used for sensitive data.

Question 1.6

Which namespace contains Kubernetes system components?

A) default
B) kube-public
C) kube-system
D) kube-node-lease

Show Answer

Answer: C) kube-system The kube-system namespace contains objects created by the Kubernetes system, such as kube-dns, kube-proxy, and other system components.

Question 1.7

What does a ReplicaSet ensure?

A) Pods are scheduled on specific nodes
B) A specified number of pod replicas are running at any given time
C) Pods have persistent storage
D) Pods can communicate with external services

Show Answer
Answer: B) A specified number of pod replicas are running at any given time A ReplicaSet's purpose is to maintain a stable set of replica Pods running at any given time.

Question 1.8

Which API version is used for Deployments?

A) v1
B) apps/v1
C) batch/v1
D) networking.k8s.io/v1

Show Answer
Answer: B) apps/v1 Deployments use the apps/v1 API group. The core v1 API is used for Pods, Services, ConfigMaps, etc.

Section 2: Container Orchestration (22%)

Question 2.1

What is the main difference between a container and a virtual machine?

A) Containers are larger than VMs
B) Containers share the host OS kernel
C) VMs are faster to start
D) Containers provide hardware-level isolation

Show Answer

Answer: B) Containers share the host OS kernel Containers share the host operating system's kernel, making them lightweight and fast to start. VMs include a full OS and provide hardware-level isolation.

Question 2.2

Which Kubernetes object ensures that all (or some) nodes run a copy of a Pod?

A) Deployment
B) StatefulSet
C) DaemonSet
D) ReplicaSet

Show Answer
Answer: C) DaemonSet A DaemonSet ensures that all (or some) nodes run a copy of a Pod. Common use cases include log collectors and monitoring agents.

Question 2.3

What is the purpose of resource requests in Kubernetes?

A) To limit the maximum resources a container can use
B) To guarantee minimum resources for scheduling
C) To autoscale pods based on usage
D) To define storage requirements

Show Answer
Answer: B) To guarantee minimum resources for scheduling Resource requests specify the minimum amount of resources a container needs. The scheduler uses this to decide which node to place the Pod on.

Question 2.4

Which QoS class is assigned to a Pod when requests equal limits for all containers?

A) BestEffort
B) Burstable
C) Guaranteed
D) Standard

Show Answer

Answer: C) Guaranteed A Pod is classified as Guaranteed when every container has memory and CPU limits that equal their requests.

Question 2.5

What mechanism allows Pods to be scheduled on nodes with specific taints?

A) Node selectors
B) Node affinity
C) Tolerations
D) Pod priority

Show Answer
Answer: C) Tolerations Tolerations are applied to Pods and allow (but do not require) the Pods to schedule onto nodes with matching taints.

Section 3: Cloud Native Architecture (16%)

Question 3.1

Which of the following is a CNCF graduated project?

A) Argo
B) Flux
C) Prometheus
D) Cilium

Show Answer
Answer: C) Prometheus Prometheus is a CNCF graduated project. Graduated projects are considered stable and production-ready with wide adoption.

Question 3.2

What does the "12-factor app" methodology primarily address?

A) Security best practices
B) Building software-as-a-service applications
C) Container image optimization
D) Network configuration

Show Answer

Answer: B) Building software-as-a-service applications The 12-factor app methodology provides best practices for building modern, scalable, maintainable software-as-a-service applications.

Question 3.3

What is a key characteristic of microservices architecture?

A) Single deployment unit
B) Shared database for all services
C) Services can be deployed independently
D) Monolithic codebase

Show Answer
Answer: C) Services can be deployed independently Microservices are independently deployable services that communicate over well-defined APIs. Each service can be developed, deployed, and scaled independently.

Question 3.4

What does a service mesh primarily provide?

A) Container runtime
B) Service-to-service communication management
C) Persistent storage
D) CI/CD pipelines

Show Answer
Answer: B) Service-to-service communication management A service mesh handles service-to-service communication, providing features like traffic management, security (mTLS), and observability.

Question 3.5

What is the sidecar pattern in Kubernetes?

A) Running multiple replicas of the same container
B) Running a helper container alongside the main application container
C) Running containers on dedicated nodes
D) Running containers with elevated privileges

Show Answer

Answer: B) Running a helper container alongside the main application container The sidecar pattern involves running a helper container in the same Pod as the main application to provide supporting functionality like logging, proxying, or configuration.

Section 4: Cloud Native Observability (8%)

Question 4.1

What are the three pillars of observability?

A) CPU, Memory, Disk
B) Logs, Metrics, Traces
C) Pods, Services, Deployments
D) Authentication, Authorization, Admission

Show Answer
Answer: B) Logs, Metrics, Traces The three pillars of observability are Logs (event records), Metrics (numeric measurements), and Traces (request flow through systems).

Question 4.2

Which Prometheus metric type can only increase?

A) Gauge
B) Counter
C) Histogram
D) Summary

Show Answer
Answer: B) Counter A Counter is a cumulative metric that can only increase (or reset to zero on restart). It's used for values like total requests or errors.

Question 4.3

What is the primary purpose of distributed tracing?

A) Storing application logs
B) Tracking requests across multiple services
C) Monitoring CPU usage
D) Managing secrets

Show Answer

Answer: B) Tracking requests across multiple services Distributed tracing tracks requests as they flow through multiple services, helping identify performance bottlenecks and errors.

Question 4.4

Which CNCF project provides a unified observability framework for metrics, logs, and traces?

A) Prometheus
B) Jaeger
C) OpenTelemetry
D) Fluentd

Show Answer
Answer: C) OpenTelemetry OpenTelemetry provides a single set of APIs, libraries, and agents to collect metrics, logs, and traces from applications.

Section 5: Cloud Native Application Delivery (8%)

Question 5.1

What is GitOps?

A) Using Git for version control
B) Using Git as the single source of truth for declarative infrastructure
C) A Git hosting service
D) A CI/CD tool

Show Answer
Answer: B) Using Git as the single source of truth for declarative infrastructure GitOps uses Git repositories as the source of truth for defining the desired state of infrastructure and applications, with automated synchronization.

Question 5.2

What is a Helm chart?

A) A monitoring dashboard
B) A package of pre-configured Kubernetes resources
C) A network diagram
D) A security policy

Show Answer

Answer: B) A package of pre-configured Kubernetes resources A Helm chart is a collection of files that describe a related set of Kubernetes resources, allowing for easy deployment and management.

Question 5.3

Which deployment strategy gradually shifts traffic from the old version to the new version?

A) Rolling update
B) Blue-green deployment
C) Canary deployment
D) Recreate deployment

Show Answer
Answer: C) Canary deployment Canary deployment gradually shifts traffic to the new version, allowing you to test with a small percentage of users before full rollout.

Question 5.4

What is the default deployment strategy in Kubernetes?

A) Recreate
B) Blue-green
C) Canary
D) Rolling update

Show Answer
Answer: D) Rolling update Rolling update is the default deployment strategy in Kubernetes. It gradually replaces old Pods with new ones, ensuring zero downtime.

Question 5.5

Which tool is a CNCF project for GitOps continuous delivery?

A) Jenkins
B) Argo CD
C) GitHub Actions
D) CircleCI

Show Answer
Answer: B) Argo CD Argo CD is a CNCF project that provides declarative, GitOps continuous delivery for Kubernetes.

Scenario-Based Questions

Scenario 1

You need to deploy an application that requires exactly one instance running on every node in your cluster for log collection.

Question: Which Kubernetes object should you use?

A) Deployment with node affinity
B) StatefulSet
C) DaemonSet
D) ReplicaSet with pod anti-affinity

Show Answer
Answer: C) DaemonSet DaemonSet ensures that a copy of a Pod runs on all (or selected) nodes. This is ideal for node-level services like log collectors, monitoring agents, or network plugins.

Scenario 2

Your application needs to maintain stable network identities and persistent storage across Pod restarts.

Question: Which Kubernetes object is most appropriate?

A) Deployment
B) StatefulSet
C) DaemonSet
D) Job

Show Answer
Answer: B) StatefulSet StatefulSet is designed for stateful applications that require stable network identities, persistent storage, and ordered deployment/scaling.

Scenario 3

You want to expose your application to external traffic and need the cloud provider to provision a load balancer.

Question: Which Service type should you use?

A) ClusterIP
B) NodePort
C) LoadBalancer
D) ExternalName

Show Answer
Answer: C) LoadBalancer LoadBalancer Service type exposes the Service externally using a cloud provider's load balancer. It automatically provisions an external load balancer.

Study Tips

1. **Understand core concepts** - Know the difference between Pods, Deployments, Services, etc.
2. **Practice with kubectl** - Hands-on experience helps solidify concepts
3. **Know the CNCF landscape** - Understand graduated, incubating, and sandbox projects
4. **Review Kubernetes architecture** - Understand control plane and node components
5. **Study the 12-factor app** - Know the principles for cloud native applications

---

← Back to KCNA Overview