**Certified Kubernetes Administrator (CKA)** exam:

### **Exam Overview**

* **Focus:** Validates the skills required to perform the responsibilities of a Kubernetes administrator in the industry.
* **Type:** Hands-on, performance-based test.
* **Format:** Online, proctored.
* **Duration:** 2 hours.
* **Domains Covered:**
  + **Cluster Architecture, Installation, & Configuration:** 25%
  + **Workloads & Scheduling:** 15%
  + **Services & Networking:** 20%
  + **Storage:** 10%
  + **Troubleshooting:** 30%

### **Preparation Guide**

1. **Topics to Study:**
   * Installing Kubernetes clusters using tools like kubeadm.
   * Managing workloads, including Deployments, Pods, and Jobs.
   * Configuring networking components such as Services, Ingress, and NetworkPolicies.
   * Debugging issues in clusters and resolving them.
   * Persistent storage setup and management【62】【64】.
2. **Practice Resources:**
   * CNCF Candidate Handbook and Curriculum Overview【63】【64】.
   * Labs and simulations available through platforms like KodeKloud and Killer.sh【64】.
   * Free introductory courses like CNCF’s *Introduction to Kubernetes*【64】.
3. **Tools to Enable Success:**
   * Set up a local Kubernetes environment with Minikube or Kind.
   * Use the official Kubernetes documentation during the exam (it’s allowed).

### **Exam Details**

* **Fee:** $395 (includes one free retake)【62】【64】.
* **Pass Mark:** Specific passing criteria are not disclosed by CNCF but aim to complete all tasks accurately within the time limit【62】【64】.

### **Sample Question**

**Scenario:** A pod is failing due to an incorrect image name. Debug and update the pod configuration to ensure it runs successfully.  
 **Command Line Tasks:**

* Check pod status: kubectl get pods.
* View logs: kubectl logs <pod-name>.
* Edit and fix deployment: kubectl edit deployment <deployment-name>.

Here are more **sample questions and areas to focus on for the CKA exam**:

### **Cluster Architecture, Installation, and Configuration (25%)**

1. **Install Kubernetes cluster using kubeadm:**
   * Set up a Kubernetes cluster with a control plane and worker nodes.
   * Example Task:
     + Install Kubernetes with kubeadm init --pod-network-cidr=192.168.0.0/16.
     + Join worker nodes using the kubeadm join command.
2. **Configure cluster networking:**
   * Install a network plugin like Calico or Flannel.
   * Example Question:

Apply a pod network plugin YAML:  
 kubectl apply -f https://docs.projectcalico.org/manifests/calico.yaml

1. **Upgrade Kubernetes version:**
   * Upgrade clusters using kubeadm and update kubelet/kubectl.

### **Workloads and Scheduling (15%)**

1. **Pod Creation and Management:**
   * Create pods using YAML files or imperative commands.

Example Task:  
 Create a pod using a YAML file:  
 apiVersion: v1

kind: Pod

metadata:

name: nginx

spec:

containers:

- name: nginx

image: nginx

Apply it:  
 kubectl apply -f pod.yaml

1. **Node Affinity and Taints:**
   * Configure node affinity rules to restrict pod placement.
   * Example Question:

Schedule a pod only on nodes with a specific label (e.g., environment: dev):  
 affinity:

nodeAffinity:

requiredDuringSchedulingIgnoredDuringExecution:

nodeSelectorTerms:

- matchExpressions:

- key: environment

operator: In

values:

- dev

1. **Job and CronJob Management:**
   * Manage batch jobs and recurring tasks.
   * Example Question:

Create a CronJob to run every 5 minutes:  
 apiVersion: batch/v1

kind: CronJob

metadata:

name: example

spec:

schedule: "\*/5 \* \* \* \*"

jobTemplate:

spec:

template:

spec:

containers:

- name: hello

image: busybox

command: ["echo", "Hello Kubernetes"]

restartPolicy: OnFailure

### **Services and Networking (20%)**

1. **Create Services:**

Example Task: Create a LoadBalancer service:  
 apiVersion: v1

kind: Service

metadata:

name: nginx-service

spec:

type: LoadBalancer

selector:

app: nginx

ports:

- protocol: TCP

port: 80

targetPort: 80

Use the command:  
 kubectl apply -f service.yaml

1. **Ingress Configuration:**
   * Create and expose applications via Ingress.
2. **Network Policies:**
   * Example Question: Write a NetworkPolicy to allow traffic only from pods in the same namespace.

### **Storage (10%)**

1. **Persistent Volumes (PV) and Persistent Volume Claims (PVC):**

Example Task: Create a PVC for dynamic provisioning:  
 apiVersion: v1

kind: PersistentVolumeClaim

metadata:

name: pvc-demo

spec:

accessModes:

- ReadWriteOnce

resources:

requests:

storage: 1Gi

Apply it:  
 kubectl apply -f pvc.yaml

1. **StorageClass Management:**
   * Use different storage classes to handle storage backends.

### **Troubleshooting (30%)**

1. **Identify Pod Issues:**

Example Question: Debug a failing pod:  
 kubectl logs <pod-name>

kubectl describe pod <pod-name>

1. **Fix Node Failures:**

Example Task: Check node health:  
 kubectl get nodes

kubectl describe node <node-name>

1. **Diagnose Cluster Networking:**
   * Example Question: Use tools like kubectl exec to ping between pods.

### **Additional Study Resources**

1. **CNCF GitHub Curriculum:** CNCF offers a detailed curriculum to prepare for the CKA exam. Review it on their GitHub page【63†source】.
2. **Practice Platforms:**
   * Killer.sh (CNCF official lab partner).
   * KodeKloud hands-on labs.

Let me know if you'd like more tailored questions or additional examples!

### **How to Register**

Register through the **Linux Foundation Training portal**. CNCF also offers wholesale pricing for bulk purchases by training organizations【62】【64】.

Let me know if you'd like additional information or guidance!