



# CKA<sup>Q&As</sup>

Certified Kubernetes Administrator (CKA) Program

## Pass Linux Foundation CKA Exam with 100% Guarantee

Free Download Real Questions & Answers **PDF** and **VCE** file from:

<https://www.pass4itsure.com/cka.html>

100% Passing Guarantee  
100% Money Back Assurance

Following Questions and Answers are all new published by Linux Foundation Official Exam Center

-  **Instant Download** After Purchase
-  **100% Money Back** Guarantee
-  **365 Days** Free Update
-  **800,000+** Satisfied Customers





## QUESTION 1

Create a pod that having 3 containers in it? (Multi-Container)

Correct Answer: Check the answer in explanation.

image=nginx, image=redis, image=consul Name nginx container as "nginx-container" Name redis container as "redis-container" Name consul container as "consul-container" Create a pod manifest file for a container and append container section for rest of the images kubectl run multi-container --generator=run-pod/v1 --image=nginx -- dry-run -o yaml > multi-container.yaml # then vim multi-container.yaml apiVersion: v1 kind: Pod metadata: labels: run: multi-container name: multi-container spec: containers:

-

image: nginx name: nginx-container

-

image: redis name: redis-container

-

image: consul name: consul-container restartPolicy: Always

---

## QUESTION 2

### SIMULATION

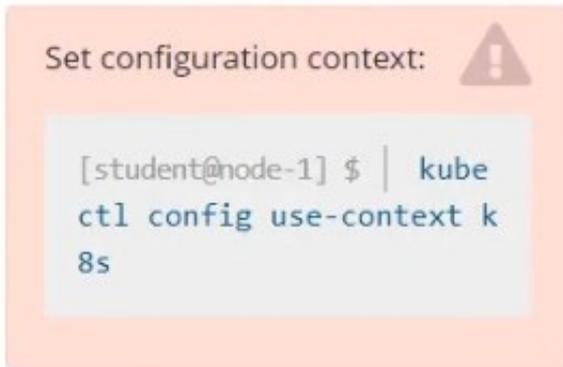
Check to see how many worker nodes are ready (not including nodes tainted NoSchedule) and write the number to /opt/KUCC00104/kucc00104.txt.

Correct Answer: Check the answer in explanation.



**QUESTION 3**

## SIMULATION



## Task

Reconfigure the existing deployment front-end and add a port specification named http exposing port 80/tcp of the existing container nginx. Create a new service named front-end-svc exposing the container port http.

Configure the new service to also expose the individual Pods via a NodePort on the nodes on which they are scheduled.

Correct Answer: Check the answer in explanation.

```
student@node-1:~$ kubectl config use-context k8s  
Switched to context "k8s".  
student@node-1:~$ kubectl get deployments.apps  
NAME           READY   UP-TO-DATE   AVAILABLE   AGE  
front-end      2/2     2            2           5h57m  
presentation   2/2     2            2           5h56m  
student@node-1:~$ kubectl edit deployments.apps front-end
```



```
❏ Please edit the object below. Lines beginning with a '#' will be ignored,
# and an empty file will abort the edit. If an error occurs while saving this file will be
# reopened with the relevant failures.
#
apiVersion: apps/v1
kind: Deployment
metadata:
  annotations:
    deployment.kubernetes.io/revision: "1"
    kubectrl.kubernetes.io/last-applied-configuration: |
      [{"apiVersion":"apps/v1","kind":"Deployment","metadata":{"annotations":{},"name":"front-end","namespace":"default"},"spec":{"replicas":2,"selector":{"matchLabels":{"app":"front-end"},"template":{"metadata":{"labels":{"app":"front-end"},"spec":{"containers":[{"image":"nginx:1.14.2","name":"nginx"}]}}}}}]
  creationTimestamp: "2022-04-25T09:24:15s"
  generation: 1
  name: front-end
  namespace: default
  resourceVersion: "3939"
  uid: 1db4fd19-6a6e-4639-a39e-25f836be0017
spec:
  progressDeadlineSeconds: 600
  replicas: 2
  revisionHistoryLimit: 10
  selector:
    matchLabels:
      app: front-end
  strategy:
    rollingUpdate:
      maxSurge: 25%
      maxUnavailable: 25%
      type: RollingUpdate
  template:
    metadata:
      creationTimestamp: null
      labels:
        app: front-end
    spec:
      containers:
      - image: nginx:1.14.2
        imagePullPolicy: IfNotPresent
        name: nginx
        ports:
        - containerPort: 80
          name: http
        resources: {}
        terminationMessagePath: /dev/termination-log
        terminationMessagePolicy: File
      dnsPolicy: ClusterFirst
      restartPolicy: Always
      schedulerName: default-scheduler
      securityContext: {}
      terminationGracePeriodSeconds: 30
status:
  availableReplicas: 2
:wc
```

```
student@node-1:~$ kubectl config use-context k8s
Switched to context "k8s".
student@node-1:~$ kubectl get deployments.apps
NAME          READY  UP-TO-DATE  AVAILABLE  AGE
front-end    2/2    2           2          5h57m
presentation 2/2    2           2          5h56m
student@node-1:~$ kubectl edit deployments.apps front-end
deployment.apps/front-end edited
student@node-1:~$ kubectl expose deployment front-end --name=front-end-svc --port=80 --type=NodePort --protocol=TCP
service/front-end-svc exposed
student@node-1:~$ kubectl describe svc front-end-svc
Name:          front-end-svc
Namespace:     default
Labels:        <none>
Annotations:   <none>
Selector:      app=front-end
Type:          NodePort
IP Family Policy: SingleStack
IP Families:   IPv4
IP:            10.107.66.230
IPs:           10.107.66.230
Port:          <unset> 80/TCP
TargetPort:    80/TCP
NodePort:      <unset> 30392/TCP
Endpoints:     10.244.1.9:80,10.244.2.8:80
Session Affinity: None
External Traffic Policy: Cluster
Events:        <none>
student@node-1:~$
```



#### QUESTION 4

Create a busybox pod and add "sleep 3600" command

Correct Answer: Check the answer in explanation.

Solution

```
kubectl run busybox --image=busybox --restart=Never -- /bin/sh -c "sleep 3600"
```

---

#### QUESTION 5

Create a busybox pod that runs the command "env" and save the output to "envpod" file

Correct Answer: Check the answer in explanation.

Solution

```
kubectl run busybox --image=busybox --restart=Never -rm -it -- env>; envpod.yaml
```

[CKA VCE Dumps](#)

[CKA Study Guide](#)

[CKA Brindumps](#)