



CKAD^{Q&As}

Certified Kubernetes Application Developer (CKAD) Program

Pass Linux Foundation CKAD Exam with 100% Guarantee

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

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

100% Passing Guarantee
100% Money Back Assurance

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

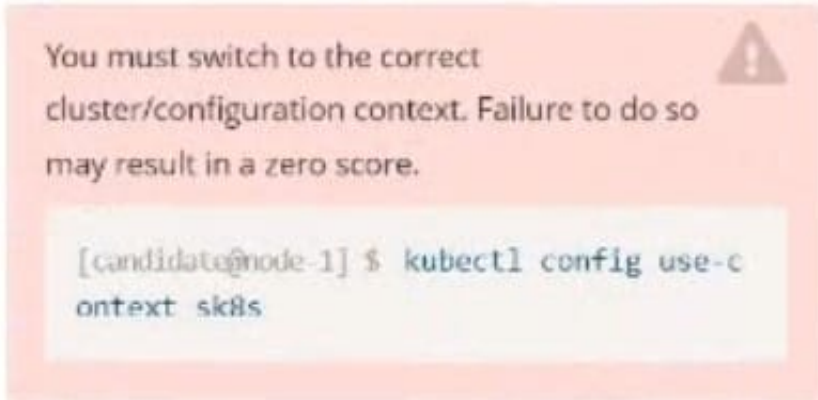
-  **Instant Download** After Purchase
-  **100% Money Back** Guarantee
-  **365 Days** Free Update
-  **800,000+** Satisfied Customers





QUESTION 1

CORRECT TEXT



Task:

Create a Pod named `nginx` resources in the existing `pod-resources` namespace.

Specify a single container using `nginx:stable` image.

Specify a resource request of 300m cpus and 1Gi of memory for the Pod's container.

A. Please check explanations

B. Place Holder

Correct Answer: A



```
candidate@node-1:~$ kubectl config use-context k8s
Switched to context "k8s".
candidate@node-1:~$ kubectl run nginx-resources -n pod-resources --image=nginx:stable --dry-run=client -o yaml > hw.yaml
candidate@node-1:~$ vim hw.yaml
```

```
File Edit View Terminal Tabs Help
apiVersion: v1
kind: Pod
metadata:
  creationTimestamp: null
  labels:
    run: nginx-resources
  name: nginx-resources
  namespace: pod-resources
spec:
  containers:
  - image: nginx:stable
    name: nginx-resources
    resources:
      requests:
        cpu: 300m
        memory: "1Gi"
```



```
candidate@node-1:~$ kubectl config use-context k8s
Switched to context "k8s".
candidate@node-1:~$ kubectl run nginx-resources -n pod-resources --image=nginx:stable --dry-run=client -o yaml > hw.yaml
candidate@node-1:~$ vim hw.yaml
candidate@node-1:~$ kubectl create -f hw.yaml
pod/nginx-resources created
candidate@node-1:~$ kubectl get pods -n pod-resources
NAME          READY   STATUS    RESTARTS   AGE
nginx-resources 1/1     Running   0           13s
candidate@node-1:~$ kubectl describe pods -n pod-resources
```

```
File Edit View Terminal Tabs Help
memory:      1Gi
Environment: <none>
Mounts:
  /var/run/secrets/kubernetes.io/serviceaccount from kube-api-access-dmx9j (ro)
Conditions:
  Type             Status
  Initialized       True
  Ready             True
  ContainersReady  True
  PodScheduled     True
Volumes:
  kube-api-access-dmx9j:
    Type:              Projected (a volume that contains injected data from multiple sources)
    TokenExpirationSeconds: 3607
    ConfigMapName:      kube-root-ca.crt
    ConfigMapOptional:  <nil>
    DownwardAPI:       true
QoS Class:           Burstable
Node-Selectors:      <none>
Tolerations:         node.kubernetes.io/not-ready:NoExecute op=Exists for 300s
                     node.kubernetes.io/unreachable:NoExecute op=Exists for 300s
Events:
  Type    Reason      Age   From          Message
  ----    -
  Normal  Scheduled  20s   default-scheduler  Successfully assigned pod-resources/nginx-resources to k8s-node-0
  Normal  Pulling    19s   kubelet        Pulling image "nginx:stable"
  Normal  Pulled     13s   kubelet        Successfully pulled image "nginx:stable" in 6.55664052s
  Normal  Created    13s   kubelet        Created container nginx-resources
  Normal  Started    12s   kubelet        Started container nginx-resources
candidate@node-1:~$ kubectl config use-context k8s
Switched to context "k8s".
candidate@node-1:~$ kubectl create deploy expose -n ckad00014 --image lfccncf/nginx:1.13.7 --dry-run=client -o yaml >
```

QUESTION 2

CORRECT TEXT



Context You are tasked to create a secret and consume the secret in a pod using environment variables as follow: Task



1.

Create a secret named another-secret with a key/value pair; key1/value4

2.

Start an nginx pod named nginx-secret using container image nginx, and add an environment variable exposing the value of the secret key key1, using COOL_VARIABLE as the name for the environment variable inside the pod

A. Please check explanations

B. Place Holder

Correct Answer: A

```
student@node-1:~$ kubectl create secret generic some-secret --from-literal=key1=value4
secret/some-secret created
student@node-1:~$ kubectl get secret
NAME                                TYPE                                DATA  AGE
default-token-4kvr5                 kubernetes.io/service-account-token  3      2d11h
some-secret                          Opaque                              1      5s
student@node-1:~$ kubectl run nginx-secret --image=nginx --dry-run=client -o yaml > nginx_secret
.yml
student@node-1:~$ vim nginx_secret.yml
```

The screenshot shows a web terminal interface with a blue header containing "Readme" and "Web Terminal" buttons, and "THE LINUX FOUNDATION" logo. The terminal displays the following YAML content:

```
apiVersion: v1
kind: Pod
metadata:
  creationTimestamp: null
  labels:
    run: nginx-secret
  name: nginx-secret
spec:
  containers:
  - image: nginx
    name: nginx-secret
    resources: {}
  dnsPolicy: ClusterFirst
  restartPolicy: Always
status: {}
```

At the bottom of the terminal, it shows the file name and size: "nginx_secret.yml" 15L, 253C. The cursor is at line 1, column 1.



```
Readme Web Terminal THE LINUX FOUNDATION

apiVersion: v1
kind: Pod
metadata:
  labels:
    run: nginx-secret
    name: nginx-secret
spec:
  containers:
  - image: nginx
    name: nginx-secret
    env:
    - name: COOL_VARIABLE
      valueFrom:
        secretKeyRef:
          name: some-secret
          key: key1
~
~
~
~
~
~
~
~
-- INSERT -- 16,20 All
```

```
Readme Web Terminal THE LINUX FOUNDATION

student@node-1:~$ kubectl get pods -n web
NAME      READY   STATUS    RESTARTS   AGE
cache     1/1     Running   0           9s
student@node-1:~$ kubectl create secret generic some-secret --from-literal=key1=value4
secret/some-secret created
student@node-1:~$ kubectl get secret
NAME                TYPE          DATA   AGE
default-token-4kvr5 kubernetes.io/service-account-token 3       2d11h
some-secret         Opaque        1       5s
student@node-1:~$ kubectl run nginx-secret --image=nginx --dry-run=client -o yaml > nginx_secret.yml
student@node-1:~$ vim nginx_secret.yml
student@node-1:~$ kubectl create -f nginx_secret.yml
pod/nginx-secret created
student@node-1:~$ kubectl get pods
NAME                READY   STATUS             RESTARTS   AGE
liveness-http      1/1     Running            0           6h38m
nginx-101           1/1     Running            0           6h39m
nginx-secret       0/1     ContainerCreating  0           4s
poller              1/1     Running            0           6h39m
student@node-1:~$ kubectl get pods
NAME                READY   STATUS    RESTARTS   AGE
liveness-http      1/1     Running   0           6h38m
nginx-101           1/1     Running   0           6h39m
nginx-secret       1/1     Running   0           8s
poller              1/1     Running   0           6h39m
student@node-1:~$
```

QUESTION 3




CORRECT TEXT

```
Set configuration context:   
  
[student@node-1] $ | kubectl config  
use-context nk8s
```

Task

You have rolled out a new pod to your infrastructure and now you need to allow it to communicate with the web and storage pods but nothing else. Given the running pod `kdsn00201 -newpod` edit it to use a network policy that will allow it to send and receive traffic only to and from the web and storage pods.

All work on this item should be conducted in the `kdsn00201` namespace. 

All required `NetworkPolicy` resources are already created and ready for use as appropriate. You should not create, modify or delete any network policies whilst completing this item. 

A. Please check explanations



B. Place Holder

Correct Answer: A

apiVersion: networking.k8s.io/v1

kind: NetworkPolicy

metadata:

name: internal-policy

namespace: default

spec:

podSelector:

matchLabels:

name: internal

policyTypes:

-Egress

-Ingress ingress:

-{} egress:

-to:

-podSelector: matchLabels:

name: mysql ports:

-protocol: TCP port: 3306

-to:

-podSelector: matchLabels: name: payroll ports:

-protocol: TCP port: 8080

-ports:

-

port: 53 protocol: UDP

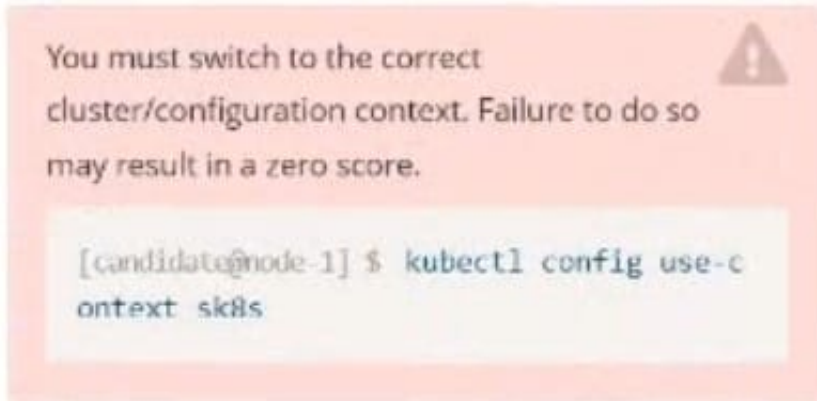
-

port: 53 protocol: TCP

QUESTION 4



CORRECT TEXT



Task:

The pod for the Deployment named nosql in the crayfish namespace fails to start because its container runs out of resources.

Update the nosql Deployment so that the Pod:

• The nosql Deployment's manifest file can be found at `~/chief-cardinal/nosql.yaml`.

A. Please check explanations

B. Place Holder

Correct Answer: A



```
candidate@node-1:~$ kubectl config use-context k8s
Switched to context "k8s".
candidate@node-1:~$ vim ~/chief-cardinal/nosql.yaml
```

```
File Edit View Terminal Tabs Help
---
apiVersion: apps/v1
kind: Deployment
metadata:
  name: nosql
  namespace: crayfish
  labels:
    app.kubernetes.io/name: nosql
    app.kubernetes.io/component: backend
spec:
  selector:
    matchLabels:
      app.kubernetes.io/name: nosql
      app.kubernetes.io/component: backend
  replicas: 1
  template:
    metadata:
      labels:
        app.kubernetes.io/name: nosql
        app.kubernetes.io/component: backend
    spec:
      containers:
        - name: mongo
          image: mongo:4.2
          args:
            - --bind_ip
            - 0.0.0.0
          ports:
            - containerPort: 27017
-- INSERT --
12,1 All
```

```
File Edit View Terminal Tabs Help
- name: mongo
  image: mongo:4.2
  args:
    - --bind_ip
    - 0.0.0.0
  ports:
    - containerPort: 27017
  resources:
    requests:
      memory: "166Mi"
    limits:
      memory: "320Mi"
:wq
```



```
File Edit View Terminal Tabs Help
To: <any> (traffic not restricted by destination)
Policy Types: Ingress, Egress

Name:          default-deny
Namespace:     ckad00018
Created on:    2022-09-24 04:27:37 +0000 UTC
Labels:       <none>
Annotations:  <none>
Spec:
  PodSelector: <none> (Allowing the specific traffic to all pods in this namespace)
  Allowing ingress traffic:
    <none> (Selected pods are isolated for ingress connectivity)
  Not affecting egress traffic
  Policy Types: Ingress
candidate@node-1:~$ kubectl label pod ckad00018-newpod -n ckad00018 web-access=true
pod/ckad00018-newpod labeled
candidate@node-1:~$ kubectl label pod ckad00018-newpod -n ckad00018 db-access=true
pod/ckad00018-newpod labeled
candidate@node-1:~$ kubectl config use-context k8s
Switched to context "k8s".
candidate@node-1:~$ vim ~/chief-cardinal/nosql.yaml
candidate@node-1:~$ vim ~/chief-cardinal/nosql.yaml
candidate@node-1:~$ kubectl apply -f ~/chief-cardinal/nosql.yaml
deployment.apps/nosql configured
candidate@node-1:~$ kubectl get pods -n crayfish
NAME                                READY   STATUS    RESTARTS   AGE
nosql-74cccf7d64-lkqlg             1/1     Running   0           3m2s
candidate@node-1:~$ kubectl get deploy -n crayfish
NAME    READY   UP-TO-DATE   AVAILABLE   AGE
nosql   1/1     1             1           7h16m
candidate@node-1:~$
```

QUESTION 5

CORRECT TEXT



Context

A web application requires a specific version of redis to be used as a cache.

Task

Create a pod with the following characteristics, and leave it running when complete:

1.

The pod must run in the web namespace.

2.

The namespace has already been created



3.

The name of the pod should be cache

4.

Use the lfcncf/redis image with the 3.2 tag

5.

Expose port 6379

A. Please check explanations

B. Place Holder

Correct Answer: A

```
student@node-1:~$ kubectl run cache --image=lfcncf/redis:3.2 --port=6379 -n web
pod/cache created
student@node-1:~$ kubectl get pods -n web
NAME      READY   STATUS             RESTARTS   AGE
cache     0/1     ContainerCreating  0           6s
student@node-1:~$ kubectl get pods -n web
NAME      READY   STATUS    RESTARTS   AGE
cache     1/1     Running   0           9s
student@node-1:~$
```

[CKAD PDF Dumps](#)

[CKAD VCE Dumps](#)

[CKAD Study Guide](#)