



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.passapply.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 new deployment for running nginx with the following parameters:

1.
Run the deployment in the kdpd00201 namespace. The namespace has already been created
2.
Name the deployment frontend and configure with 4 replicas
3.
Configure the pod with a container image of lfcncf/nginx:1.13.7
4.
Set an environment variable of NGINX__PORT=8080 and also expose that port for the container above

A. Please check explanations

B. Place Holder

Correct Answer: A



```
student@node-1:~$ kubectl create deployment api --image=lfcncf/nginx:1.13.7-alpine --replicas=4  
-n kdpd00201 --dry-run=client -o yaml > nginx_deployment.yml  
student@node-1:~$ vim nginx_deployment.yml
```

```
apiVersion: apps/v1  
kind: Deployment  
metadata:  
  creationTimestamp: null  
  labels:  
    app: api  
  name: api  
  namespace: kdpd00201  
spec:  
  replicas: 4  
  selector:  
    matchLabels:  
      app: api  
  strategy: {}  
  template:  
    metadata:  
      creationTimestamp: null  
      labels:  
        app: api  
    spec:  
      containers:  
        - image: lfcncf/nginx:1.13.7-alpine  
          name: nginx  
          resources: {}  
status: {}  
~  
"nginx_deployment.yml" 25L, 421C 4,1 All
```

```
apiVersion: apps/v1  
kind: Deployment  
metadata:  
  labels:  
    app: api  
  name: api  
  namespace: kdpd00201  
spec:  
  replicas: 4  
  selector:  
    matchLabels:  
      app: api  
  template:  
    metadata:  
      labels:  
        app: api  
    spec:  
      containers:  
        - image: lfcncf/nginx:1.13.7-alpine  
          name: nginx  
          ports:  
            - containerPort: 8080  
          env:  
            - name: NGINX_PORT  
              value: "8080"
```

```
student@node-1:~$ kubectl create deployment api --image=lfcncf/nginx:1.13.7-alpine --replicas=4  
-n kdpd00201 --dry-run=client -o yaml > nginx_deployment.yml  
student@node-1:~$ vim nginx_deployment.yml  
student@node-1:~$ kubectl create nginx_deployment.yml  
Error: must specify one of -f and -k  
  
error: unknown command "nginx_deployment.yml"  
See 'kubectl create -h' for help and examples  
student@node-1:~$ kubectl create -f nginx_deployment.yml  
error: error validating "nginx_deployment.yml": error validating data: ValidationError(Deployment.spec.template.spec): unknown field "env" in io.k8s.api.core.v1.PodSpec; if you choose to ignore these errors, turn validation off with --validate=false  
student@node-1:~$ vim nginx_deployment.yml  
student@node-1:~$ kubectl create -f nginx_deployment.yml  
deployment.apps/api created  
student@node-1:~$ kubectl get pods -n kdpd00201  
NAME READY STATUS RESTARTS AGE  
api-745677f7dc-7hnmv 1/1 Running 0 13s  
api-745677f7dc-9q5vp 1/1 Running 0 13s  
api-745677f7dc-fd4gk 1/1 Running 0 13s  
api-745677f7dc-mbnpk 1/1 Running 0 13s  
student@node-1:~$
```



QUESTION 2

CORRECT TEXT



Context

You are tasked to create a ConfigMap and consume the ConfigMap in a pod using a volume mount.

Task

Please complete the following:

1.

Create a ConfigMap named another-config containing the key/value pair: key4/value3

2.

start a pod named nginx-configmap containing a single container using the nginx image, and mount the key you just created into the pod under directory /also/a/path

A. Please check explanations

B. Place Holder

Correct Answer: A

```
student@node-1:~$ kubectl create configmap another-config --from-literal=key4=value3
configmap/another-config created
student@node-1:~$ kubectl get configmap
NAME          DATA   AGE
another-config  1       5s
student@node-1:~$ kubectl run nginx-configmap --image=nginx --dry-run=client -o yaml > nginx_configmap.yml
student@node-1:~$ vim nginx_configmap.yml ^C
student@node-1:~$ mv nginx_configmap.yml nginx_configmap.yml
student@node-1:~$ vim nginx_co
```



```

Readme Web Terminal THE LINUX FOUNDATION
apiVersion: v1
kind: Pod
metadata:
  creationTimestamp: null
  labels:
    run: nginx-configmap
  name: nginx-configmap
spec:
  containers:
  - image: nginx
    name: nginx-configmap
    resources: {}
  dnsPolicy: ClusterFirst
  restartPolicy: Always
status: {}
"nginx_configmap.yml" 15L, 262C 1,1 All

```

```

Readme Web Terminal THE LINUX FOUNDATION
apiVersion: v1
kind: Pod
metadata:
  labels:
    run: nginx-configmap
  name: nginx-configmap
spec:
  containers:
  - image: nginx
    name: nginx-configmap
    volumeMounts:
    - name: myvol
      mountPath: /also/a/path
  volumes:
  - name: myvol
    configMap:
      name: another-config
13,6 All

```

```

student@node-1:~$ kubectl create configmap another-config --from-literal=key4=value3
configmap/another-config created
student@node-1:~$ kubectl get configmap
NAME      DATA   AGE
another-config  1       5s
student@node-1:~$ kubectl run nginx-configmap --image=nginx --dry-run=client -o yaml > nginx_conf
igmap.yml
student@node-1:~$ vim nginx_configmap.yml ^c
student@node-1:~$ mv nginx_configmap.yml nginx_configmap.yml
student@node-1:~$ vim nginx_configmap.yml
student@node-1:~$

```

```

student@node-1:~$ kubectl run nginx-configmap --image=nginx --dry-run=client -o yaml > nginx_conf
igmap.yml
student@node-1:~$ vim nginx_configmap.yml ^c
student@node-1:~$ mv nginx_configmap.yml nginx_configmap.yml
student@node-1:~$ vim nginx_configmap.yml
student@node-1:~$ kubectl create f nginx_configmap.yml
Error: must specify one of -f and -k

error: unknown command "f nginx_configmap.yml"
See 'kubectl create -h' for help and examples
student@node-1:~$ kubectl create -f nginx_configmap.yml
error: error validating "nginx_configmap.yml": error validating data: ValidationError(Pod.spec.c
ontainers[1]): unknown field "mountPath" in io.k8s.api.core.v1.Container; if you choose to ignor
e these errors, turn validation off with --validate=false
student@node-1:~$ vim nginx_configmap.yml

```

```

Readme Web Terminal THE LINUX FOUNDATION
student@node-1:~$ kubectl create f nginx_configmap.yml
Error: must specify one of -f and -k

error: unknown command "f nginx_configmap.yml"
See 'kubectl create -h' for help and examples
student@node-1:~$ kubectl create -f nginx_configmap.yml
error: error validating "nginx_configmap.yml": error validating data: ValidationError(Pod.spec.c
ontainers[1]): unknown field "mountPath" in io.k8s.api.core.v1.Container; if you choose to ignor
e these errors, turn validation off with --validate=false
student@node-1:~$ vim nginx_configmap.yml
student@node-1:~$ kubectl create -f nginx_configmap.yml
pod/nginx-configmap created
student@node-1:~$ kubectl get pods
NAME          READY   STATUS    RESTARTS   AGE
liveness-http  1/1     Running   0           6h44m
nginx-101     1/1     Running   0           6h45m
nginx-configmap 0/1     ContainerCreating 0           5s
nginx-secret  1/1     Running   0           3m39s
poller       1/1     Running   0           6h44m
student@node-1:~$ kubectl get pods
NAME          READY   STATUS    RESTARTS   AGE
liveness-http  1/1     Running   0           6h44m
nginx-101     1/1     Running   0           6h45m
nginx-configmap 1/1     Running   0           8s
nginx-secret  1/1     Running   0           5m42s
poller       1/1     Running   0           6h45m
student@node-1:~$

```



QUESTION 3

CORRECT TEXT



Context

A project that you are working on has a requirement for persistent data to be available.

Task

To facilitate this, perform the following tasks:

1.

Create a file on node sk8s-node-0 at /opt/KDSP00101/data/index.html with the content Acct=Finance

2.

Create a PersistentVolume named task-pv-volume using hostPath and allocate 1Gi to it, specifying that the volume is at /opt/KDSP00101/data on the cluster's node.

The configuration should specify the access mode of ReadWriteOnce. It should define the StorageClass name exam for the PersistentVolume, which will be used to bind PersistentVolumeClaim requests to this PersistentVolume.

1.

Create a PersistentVolumeClaim named task-pv-claim that requests a volume of at least 100Mi and specifies an access mode of ReadWriteOnce

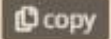
2.

Create a pod that uses the PersistentVolumeClaim as a volume with a label app: my-storage-app mounting the resulting volume to a mountPath /usr/share/nginx/html inside the pod



You can access sk8s-node-0 by  issuing the following command:

```
[student@node-1] $ | ssh sk8s-node-0
```

Ensure that you return to the base node (with hostname node-1) once you have completed your work on sk8s-node-0 

A. Please check explanations

B. Place Holder

Correct Answer: A



```
Readme Web Terminal THE LINUX FOUNDATION
student@node-1:~$ kubectl config use-context sk8s
Switched to context "sk8s".
student@node-1:~$
```

```
Readme Web Terminal THE LINUX FOUNDATION
* Documentation: https://help.ubuntu.com
* Management: https://landscape.canonical.com
* Support: https://ubuntu.com/advantage

System information as of Fri Oct 9 08:52:09 UTC 2020

System load: 2.02          Users logged in: 0
Usage of /: 10.3% of 242.29GB IP address for eth0: 10.250.3.115
Memory usage: 2%          IP address for docker0: 172.17.0.1
Swap usage: 0%            IP address for cni0: 10.244.1.1
Processes: 38

* Kubernetes 1.19 is out! Get it in one command with:

  sudo snap install microk8s --channel=1.19 --classic

https://microk8s.io/ has docs and details.

7 packages can be updated.
1 update is a security update.

New release '20.04.1 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

student@sk8s-node-0:~$
```

```
Readme Web Terminal THE LINUX FOUNDATION
student@sk8s-node-0:~$ echo 'Acct=Finance' > /opt/KDSP00101/data/index.html
student@sk8s-node-0:~$ vim pv.yml
```




```

THE LINUX FOUNDATION
Web Terminal
-- INSERT --
0,1 All

```

```

THE LINUX FOUNDATION
Web Terminal
apiVersion: v1
kind: PersistentVolume
metadata:
  name: task-pv-volume
spec:
  capacity:
    storage: 1Gi
  accessModes:
    - ReadWriteOnce
  storageClassName: storage
  hostPath:
    path: /opt/KDSP00101/data
    type: Directory

```

```

THE LINUX FOUNDATION
Web Terminal
apiVersion: v1
kind: PersistentVolumeClaim
metadata:
  name: task-pv-claim
spec:
  accessModes:
    - ReadWriteOnce
  resources:
    requests:
      storage: 100Mi
  storageClassName: storage

```

```

student@sk8a-node-01~$ kubectl create -f pv.yml
persistentvolume/task-pv-volume created
student@sk8a-node-01~$ kubectl create -f pvc.yml
persistentvolumeclaim/task-pv-claim created
student@sk8a-node-01~$ kubectl get pv
NAME             CAPACITY  ACCESS MODES  RECLAIM POLICY  STATUS  CLAIM                STORAGECLASS  AGE
task-pv-volume   1Gi       RWO           Retain          Bound   default/task-pv-claim  storage        9s
student@sk8a-node-01~$ kubectl get pvc
NAME             STATUS  VOLUME             CAPACITY  ACCESS MODES  STORAGECLASS  AGE
task-pv-claim   Bound   task-pv-volume     1Gi       RWO           storage        9s
student@sk8a-node-01~$ vim pod.yml

```

```

THE LINUX FOUNDATION
Web Terminal
apiVersion: v1
kind: Pod
metadata:
  name: mypod
  labels:
    app: my-storage-app
spec:
  containers:
    - name: myfrontend
      image: nginx
      volumeMounts:
        - mountPath: "/usr/share/nginx/html"
          name: mypod
      volume:
        - name: mypod
          persistentVolumeClaim:
            claimName: task-pv-claim

```

```

student@sk8a-node-01~$ kubectl create -f pod.yml
pod/mypod created
student@sk8a-node-01~$ kubectl get

```

```

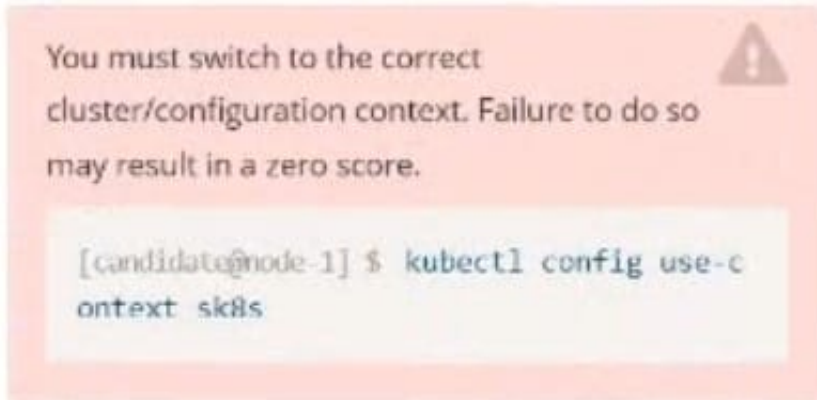
THE LINUX FOUNDATION
Web Terminal
student@sk8a-node-01~$ kubectl get pods
NAME    READY   STATUS    RESTARTS   AGE
mypod   0/1     ContainerCreating   0          4s
student@sk8a-node-01~$ kubectl get pods
NAME    READY   STATUS    RESTARTS   AGE
mypod   0/1     ContainerCreating   0          8s
student@sk8a-node-01~$ kubectl get pods
NAME    READY   STATUS    RESTARTS   AGE
mypod   1/1     Running   0          10s
student@sk8a-node-01~$ logout
Connection to 10.250.3.115 closed.
student@node-1~$

```



QUESTION 4

CORRECT TEXT



Task:

A pod within the Deployment named buffalo-deployment and in namespace gorilla is logging errors.

Look at the logs identify errors messages.

Find errors, including User "system:serviceaccount:gorilla:default" cannot list resource "deployment" [...] in the namespace "gorilla"

The buffalo-deployment `S manifest can be found at -/prompt/escargot/buffalo- deployment.yaml

A. Please check explanations

B. Place Holder

Correct Answer: A



```
File Edit View Terminal Tabs Help
deployment.apps/backend-deployment configured
candidate@node-1:~$ kubectl get pods -n staging
NAME                                READY   STATUS    RESTARTS   AGE
backend-deployment-59d449b99d-cxct6 1/1     Running   0           20s
backend-deployment-59d449b99d-h2zjq 0/1     Running   0           9s
backend-deployment-78976f74f5-b8c85 1/1     Running   0           6h40m
backend-deployment-78976f74f5-flfsj 1/1     Running   0           6h40m
candidate@node-1:~$ kubectl get deploy -n staging
NAME                READY   UP-TO-DATE   AVAILABLE   AGE
backend-deployment 3/3     3             3           6h40m
candidate@node-1:~$ kubectl get deploy -n staging
NAME                READY   UP-TO-DATE   AVAILABLE   AGE
backend-deployment 3/3     3             3           6h41m
candidate@node-1:~$ vim ~/spicy-pikachu/backend-deployment.yaml
candidate@node-1:~$ kubectl config use-context k8s
Switched to context "k8s".
candidate@node-1:~$ kubectl set serviceaccount deploy app-1 app -n frontend
deployment.apps/app-1 serviceaccount updated
candidate@node-1:~$ kubectl config use-context k8s
Switched to context "k8s".
candidate@node-1:~$ vim ~/prompt-escargot/buffalo-deployment.yaml
candidate@node-1:~$ vim ~/prompt-escargot/buffalo-deployment.yaml
candidate@node-1:~$ kubectl apply -f ~/prompt-escargot/buffalo-deployment.yaml
deployment.apps/buffalo-deployment configured
candidate@node-1:~$ kubectl get pods -n gorilla
NAME                                READY   STATUS              RESTARTS   AGE
buffalo-deployment-776844df7f-r5fsb 1/1     Running             0           5h38m
buffalo-deployment-859898c6f5-zx5gj 0/1     ContainerCreating  0           8s
candidate@node-1:~$ kubectl get deploy -n gorilla
NAME                READY   UP-TO-DATE   AVAILABLE   AGE
buffalo-deployment 1/1     1             1           6h38m
candidate@node-1:~$
```

QUESTION 5

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
```

```
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: {}
```

"nginx_secret.yml" 15L, 253C 1,1 All



```
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.yaml
student@node-1:~$ vim nginx_secret.yaml
student@node-1:~$ kubectl create -f nginx_secret.yaml
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:~$
```

[Latest CKAD Dumps](#)

[CKAD PDF Dumps](#)

[CKAD Exam Questions](#)