

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.pass2lead.com/ckad.html>

100% Passing Guarantee
100% Money Back Assurance

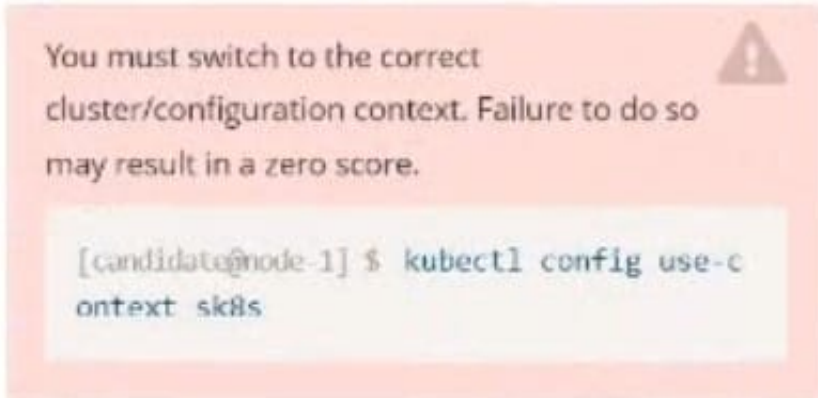
Following Questions and Answers are all new published by Linux Foundation Official Exam Center

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

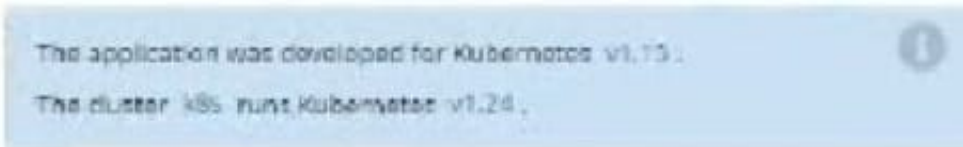


QUESTION 1

CORRECT TEXT



Task:



- 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 -/credible-mite/www.yaml
```

```
File Edit View Terminal Tabs Help
apiVersion: apps/v1
kind: Deployment
metadata:
  name: www-deployment
  namespace: cobra
spec:
  replicas: 3
  selector:
    matchLabels:
      app: nginx
template:
  metadata:
    labels:
      app: nginx
  spec:
    containers:
      - name: nginx
        image: "nginx:stable"
        ports:
          - containerPort: 80
        volumeMounts:
          - mountPath: /var/log/nginx
            name: logs
        env:
          - name: NGINX_ENTRYPOINT_QUIET_LOGS
            value: "1"
    volumes:
      - name: logs
        emptyDir: {}
~
:wq
```

```
File Edit View Terminal Tabs Help
deployment.apps/expose created
candidate@node-1:~$ kubectl get pods -n ckad00014
NAME                READY   STATUS              RESTARTS   AGE
expose-85dd99d4d9-25675  0/1    ContainerCreating   0           6s
expose-85dd99d4d9-4fhcc  0/1    ContainerCreating   0           6s
expose-85dd99d4d9-fld7j  0/1    ContainerCreating   0           6s
expose-85dd99d4d9-tt6rm  0/1    ContainerCreating   0           6s
expose-85dd99d4d9-vjd8b  0/1    ContainerCreating   0           6s
expose-85dd99d4d9-vtzpq  0/1    ContainerCreating   0           6s
candidate@node-1:~$ kubectl get deploy -n ckad00014
NAME    READY   UP-TO-DATE   AVAILABLE   AGE
expose  6/6     6             6           15s
candidate@node-1:~$ kubectl config use-context k8s
Switched to context "k8s".
candidate@node-1:~$ vim ~/credible-mite/www.yaml
candidate@node-1:~$ vim ~/credible-mite/www.yaml
candidate@node-1:~$ kubectl apply -f ~/credible-mite/www.yaml
deployment.apps/www-deployment created
candidate@node-1:~$ kubectl get pods -n cobra
NAME                READY   STATUS              RESTARTS   AGE
www-deployment-d899c6b49-d6ccg  1/1    Running             0           6s
www-deployment-d899c6b49-f796l  0/1    ContainerCreating   0           6s
www-deployment-d899c6b49-ztfcw  0/1    ContainerCreating   0           6s
candidate@node-1:~$ kubectl get deploy -n cobra
NAME    READY   UP-TO-DATE   AVAILABLE   AGE
www-deployment  3/3     3             3           11s
candidate@node-1:~$ kubectl get pods -n cobra
NAME                READY   STATUS              RESTARTS   AGE
www-deployment-d899c6b49-d6ccg  1/1    Running             0           14s
www-deployment-d899c6b49-f796l  1/1    Running             0           14s
www-deployment-d899c6b49-ztfcw  1/1    Running             0           14s
candidate@node-1:~$
```

QUESTION 2

CORRECT TEXT



Context

You sometimes need to observe a pod's logs, and write those logs to a file for further analysis.

Task

Please complete the following;

1.
Deploy the counter pod to the cluster using the provided YAMLSpec file at /opt/KDOB00201/counter.yaml
- 2.

Retrieve all currently available application logs from the running pod and store them in the file /opt/KDOB00201/log_Output.txt, which has already been created

- A. Please check explanations
- B. Place Holder

Correct Answer: A

```
student@node-1:~$ kubectl create -f /opt/KDOB00201/counter.yaml
pod/counter created
student@node-1:~$ kubectl get pods
NAME          READY   STATUS    RESTARTS   AGE
counter       1/1     Running   0           10s
liveness-http 1/1     Running   0           6h45m
nginx-101     1/1     Running   0           6h46m
nginx-configmap 1/1     Running   0           107s
nginx-secret  1/1     Running   0           7m21s
poller        1/1     Running   0           6h46m
student@node-1:~$ kubectl logs counter
1: 2b305101817ae25ca60ae46510fb6d11
2: 3648cf2eae95ab680dba8f195f891af4
3: 65c8bbd4dbf70bf81f2a0984a3a44ede
4: 40d3a9c8e46f5533bb4828fbc5c8d038
5: 390442d2530a90c3602901e3fe999ac8
6: b71d95187417e139effb33af77681040
7: 66a8e55a6491e756d2d0549ad6ab90a7
8: ff2b3d583b64125d2f9129c443bb37ff
9: b6c6a12b6e77944ed8baaaf6c242dae4
10: bfcc9a894a0604fc4b814b37d0a200a4
student@node-1:~$ kubectl logs counter > /opt/KDOB00201/log_output.txt
student@node-1:~$
```

```
student@node-1:~$ kubectl logs counter > /opt/KDOB00201/log_output.txt
student@node-1:~$ kubectl logs counter > /opt/KDOB00201/log_output.txt
student@node-1:~$ cd /opt/KDOB00201/log_output.txt
```

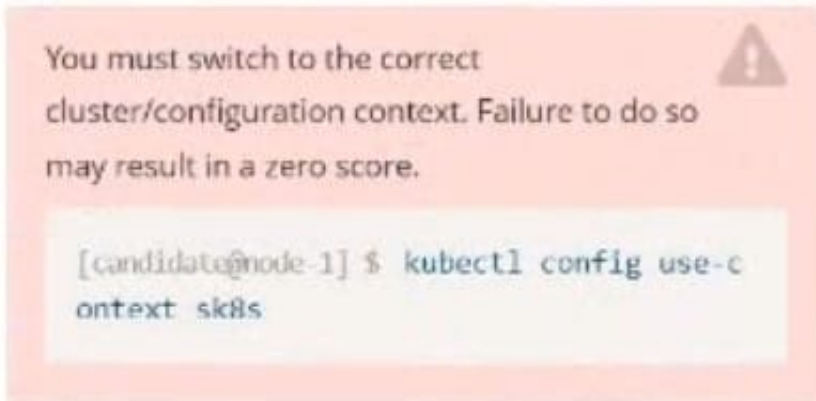
Readme Web Terminal

THE LINUX FOUNDATION

```
student@node-1:~$ kubectl logs counter > /opt/KDOB00201/log_output.txt
student@node-1:~$ cat /opt/KDOB00201/log_output.txt
1: 2b305101817ae25ca60ae46510fb6d11
2: 3648cf2eae95ab680dba8f195f891af4
3: 65c8bbd4dbf70bf81f2a0984a3a44ede
4: 40d3a9c8e46f5533bb4828fbc5c8d038
5: 390442d2530a90c3602901e3fe999ac8
6: b71d95187417e139effb33af77681040
7: 66a8e55a6491e756d2d0549ad6ab90a7
8: ff2b3d583b64125d2f9129c443bb37ff
9: b6c6a12b6e77944ed8baaaf6c242dae4
10: bfcc9a894a0604fc4b814b37d0a200a4
11: 5493cd16a1790a5fb9512b0c9d4c5dd1
12: 03f169e93e6143438e6dfe4ecb3cc9ed
13: 764b37fe611373c42d0b47154041f6eb
14: 1a56fbc1896b0ee6394136166281839e
15: ecc492eb17715de090c47345a98d98d3
16: 7974a6bec0fb44b6b8bbfc71aa3fbc74
17: 9ae01bef01748b12cc9f97a5f9f72cd6
18: 23fb22ee34d4272e4c9e005f1774515f
19: ec7e1a5d314da9a0ad45d53be5a7acae
20: 0bccdd8ee02cd42029e8162cd1c1197c
21: d6851ea43546216b95bcb81ced997102
22: 7ed9a38ea8bf0d86206569481442af44
23: 29b8416ddc63dbfcb987ab3c8198e9fe
24: 1f2062001df51a108ab25010f506716f
student@node-1:~$
```

QUESTION 3

CORRECT TEXT



Task:

Modify the existing Deployment named broker-deployment running in namespace quetzal so that its containers.

The broker-deployment is manifest file can be found at:

```
~/daring_mocasin/broker-deployment.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
```

```
File Edit View Terminal Tabs Help
```

```
containers:
- name: broker
  image: redis:alpine
  ports:
  - containerPort: 6379
  securityContext:
    runAsUser: 30000
    privileged: false
```

```
:wq
```

```
candidate@node-1:~$ kubectl config use-context k8s
Switched to context "k8s".
candidate@node-1:~$ vim ~/daring-moccasin/broker-deployment.yaml
candidate@node-1:~$ kubectl apply -f ~/daring-moccasin/broker-deployment.yaml
deployment.apps/broker-deployment configured
candidate@node-1:~$ kubectl get pods -n quetzal
NAME                                READY   STATUS    RESTARTS   AGE
broker-deployment-65446d6d94-868p6  1/1     Running   0           30s
broker-deployment-65446d6d94-8dn7l  1/1     Running   0           32s
broker-deployment-65446d6d94-p4h4l  1/1     Running   0           31s
candidate@node-1:~$ kubectl get deploy -n quetzal
NAME                READY   UP-TO-DATE   AVAILABLE   AGE
broker-deployment  3/3     3             3           7h3m
candidate@node-1:~$
```

QUESTION 4

CORRECT TEXT



Context

A user has reported an application is unreachable due to a failing livenessProbe .

Task

Perform the following tasks:

Find the broken pod and store its name and namespace to /opt/KDOB00401/broken.txt in the format:



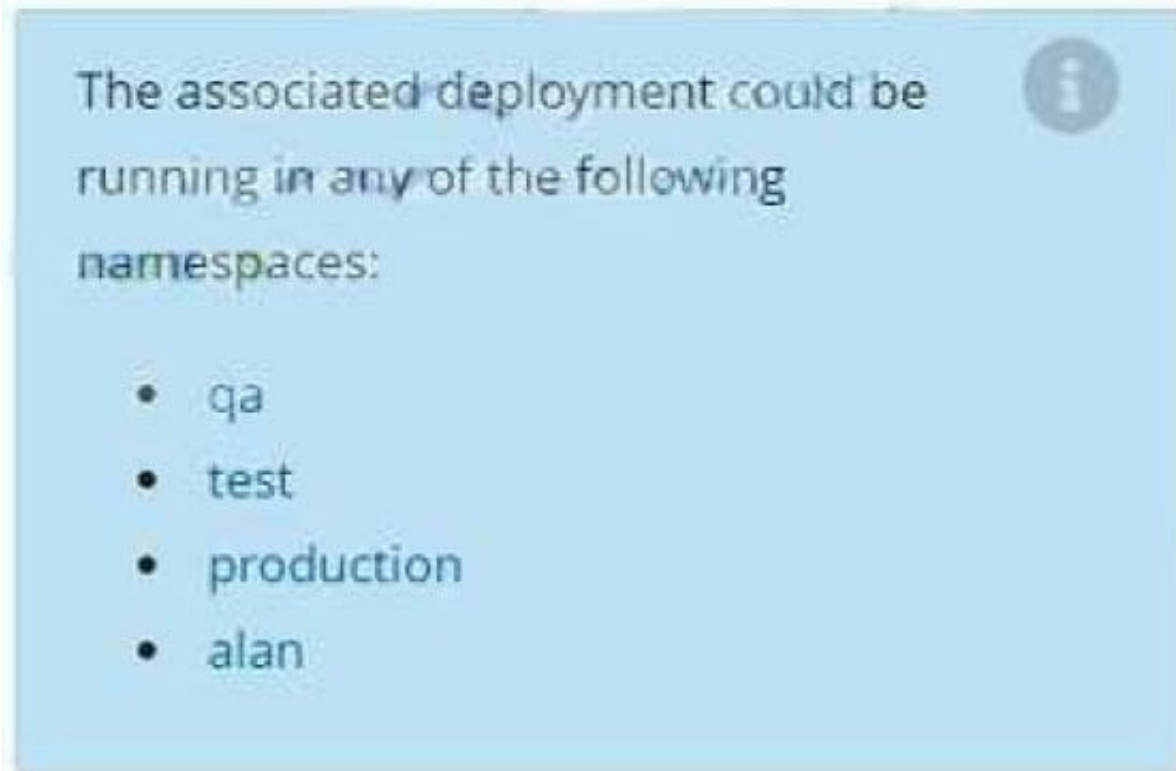
The output file has already been created

1.

Store the associated error events to a file /opt/KDOB00401/error.txt, The output file has already been created. You will need to use the -o wide output specifier with your command

2.

Fix the issue.



A. Please check explanations

B. Place Holder

Correct Answer: A

Create the Pod: `kubectl create -f http://k8s.io/docs/tasks/configure-pod-container/exec-liveness.yaml` Within 30 seconds, view the Pod events: `kubectl describe pod liveness-exec` The output indicates that no liveness probes have failed yet:

FirstSeen	LastSeen	Count	From	SubobjectPath	Type	Reason	Message
-----	24s	24s	1	{default-scheduler}	Normal	Scheduled	Successfully assigned liveness-exec to worker0
-----	23s	23s	1	{kubelet worker0}	Normal	Pulling	pulling image "gcr.io/google_containers/busybox"
-----	23s	23s	1	{kubelet worker0}	Normal	Pulled	Successfully pulled image "gcr.io/google_containers/busybox"
-----	23s	23s	1	{kubelet worker0}	Normal	Created	Created container with docker id 86849c15382e; Security:[seccomp=unconfined]
-----	23s	23s	1	{kubelet worker0}	Normal	Started	Started container with docker id 86849c15382e

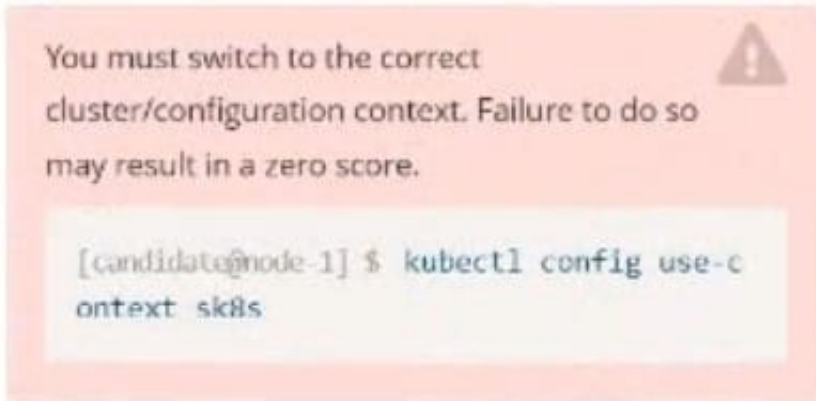
After 35 seconds, view the Pod events again: `kubectl describe pod liveness-exec` At the bottom of the output, there are messages indicating that the liveness probes have failed, and the containers have been killed and recreated.

FirstSeen	LastSeen	Count	From	SubobjectPath	Type	Reason	Message
-----	37s	37s	1	{default-scheduler}	Normal	Scheduled	Successfully assigned liveness-exec to worker0
-----	36s	36s	1	{kubelet worker0}	Normal	Pulling	pulling image "gcr.io/google_containers/busybox"
-----	36s	36s	1	{kubelet worker0}	Normal	Pulled	Successfully pulled image "gcr.io/google_containers/busybox"
-----	36s	36s	1	{kubelet worker0}	Normal	Created	Created container with docker id 86849c15382e; Security:[seccomp=unconfined]
-----	36s	36s	1	{kubelet worker0}	Normal	Started	Started container with docker id 86849c15382e
-----	2s	2s	1	{kubelet worker0}	Warning	Unhealthy	Liveness probe failed: cat: can't open \"/tmp/healthy\": No such file or directory

Wait another 30 seconds, and verify that the Container has been restarted: `kubectl get pod liveness-exec` The output shows that RESTARTS has been incremented: NAME READY STATUS RESTARTS AGE liveness-exec 1/1 Running 1 m

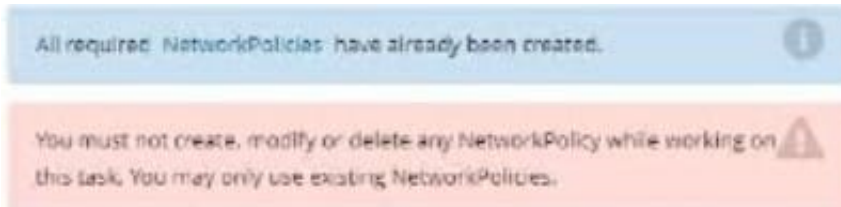
QUESTION 5

CORRECT TEXT



Task:

Update the Pod ckad00018-newpod in the ckad00018 namespace to use a NetworkPolicy allowing the Pod to send and receive traffic only to and from the pods web and db



A. Please check explanations

B. Place Holder

Correct Answer: A



```
terminal: candidate@node-1: ~
File Edit View Terminal Tabs Help
Name: all-access
Namespace: ckad00018
Created on: 2022-09-24 04:27:37 +0000 UTC
Labels: <none>
Annotations: <none>
Spec:
  PodSelector: all-access=true
  Allowing ingress traffic:
    To Port: <any> (traffic allowed to all ports)
    From: <any> (traffic not restricted by source)
  Allowing egress traffic:
    To Port: <any> (traffic allowed to all ports)
    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:~$
```

[Latest CKAD Dumps](#)

[CKAD PDF Dumps](#)

[CKAD Braindumps](#)