CKA Q&As

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.pass2lead.com/cka.html

100% Passing Guarantee
100% Money Back Assurance

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

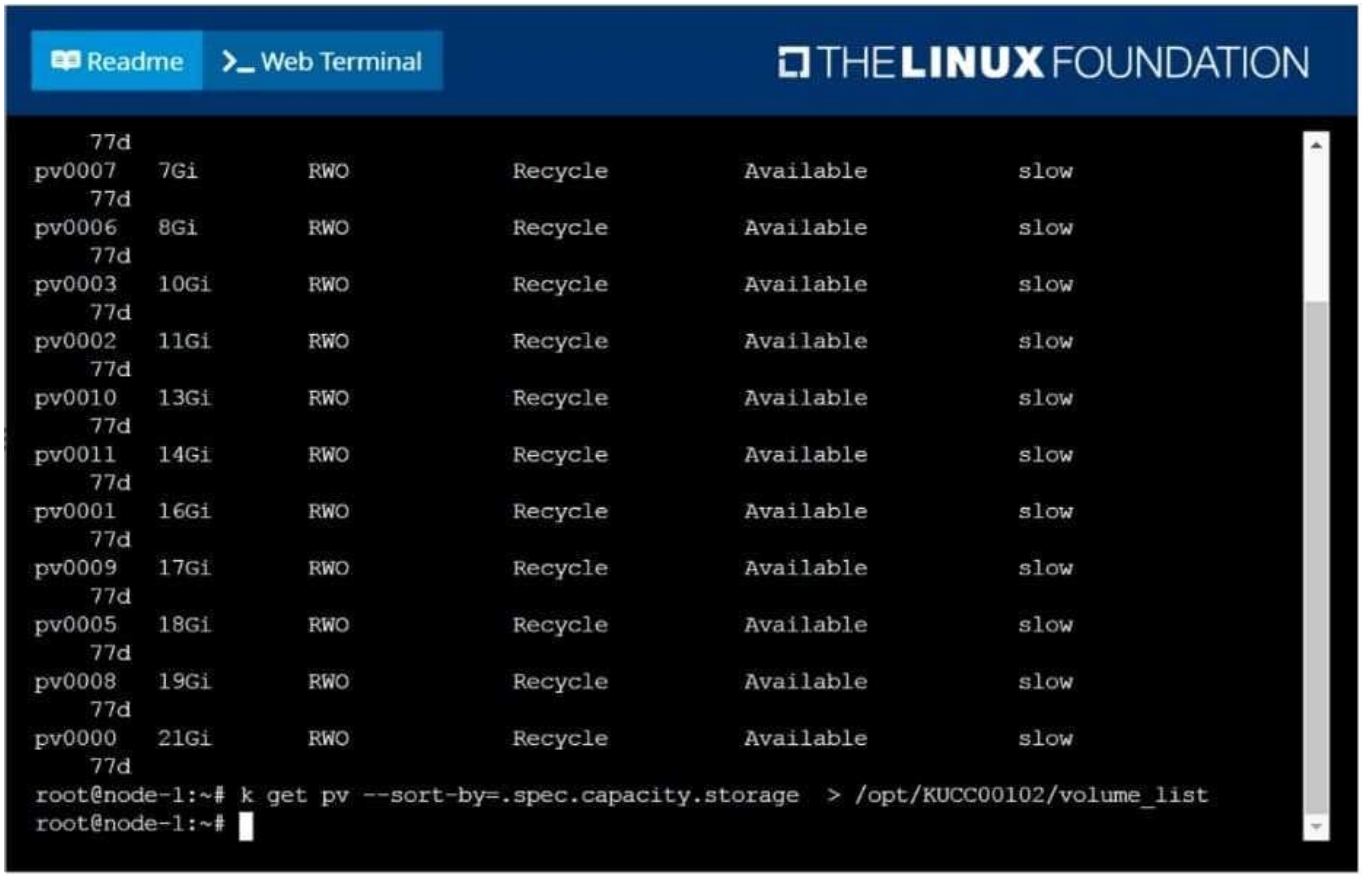
QUESTION 1

SIMULATION

List all persistent volumes sorted by capacity, saving the full kubectl output to /opt/KUCC00102/volume_list. Use kubectl \\'s own functionality for sorting the output, and do not manipulate it any further.

Correct Answer: Check the answer in explanation.

Solution



QUESTION 2

Check the image version in pod without the describe command

Correct Answer: Check the answer in explanation.

kubectl get po nginx -o jsonpath=\\'{.spec.containers[].image}{"\n"}\\'

QUESTION 3

Create a nginx pod with label env=test in engineering namespace .

Correct Answer: Check the answer in explanation.

kubectl run nginx --image=nginx --restart=Never --labels=env=test --namespace=engineering --dry- run -o yaml > nginx-pod.yaml kubectl run nginx --image=nginx --restart=Never --labels=env=test --namespace=engineering --dry- run -o yaml | kubectl create -n engineering -f YAML File:

apiVersion: v1 kind: Pod metadata: name: nginx namespace: engineering labels: env: test spec: containers:

-name: nginx image: nginx imagePullPolicy: IfNotPresent restartPolicy: Never

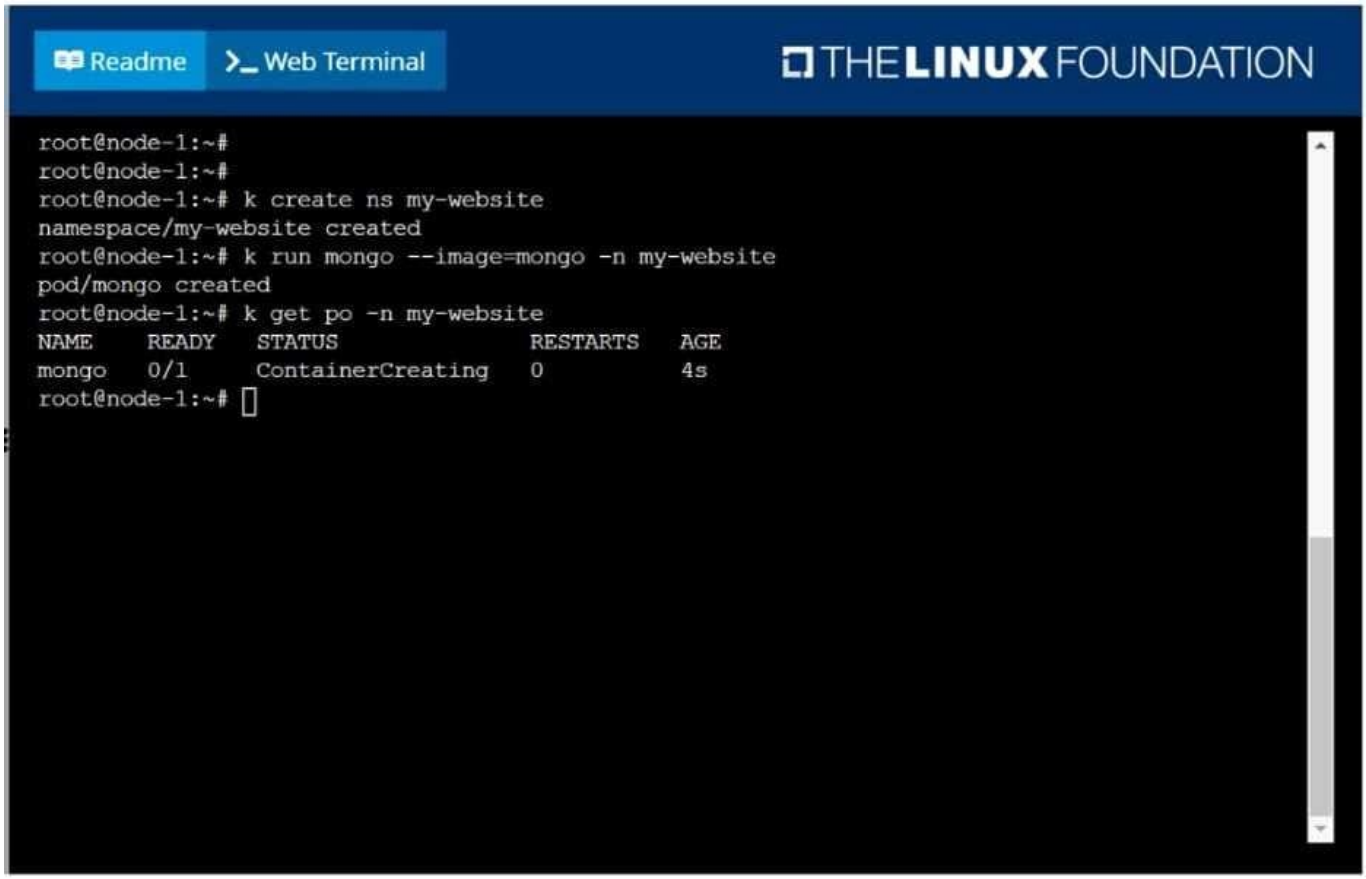
kubectl create -f nginx-pod.yaml

QUESTION 4

SIMULATION Create a pod as follows: Name: mongo Using Image: mongo In a new Kubernetes namespace named: my-website

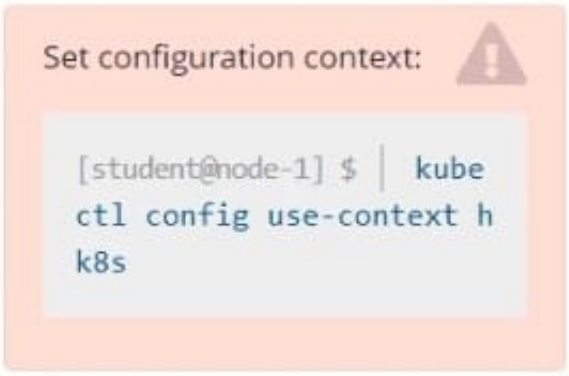
Correct Answer: Check the answer in explanation.

Solution



QUESTION 5

CORRECT TEXT



Task

Create a persistent volume with name app-data , of capacity 1Gi and access mode ReadOnlyMany. The type of volume is hostPath and its location is /srv/app-data .

Correct Answer:

#vi pv.yaml apiVersion: v1 kind: PersistentVolume metadata: name: app-config spec: capacity: storage: 1Gi accessModes:

-ReadOnlyMany hostPath: path: /srv/app-config # kubectl create -f pv.yaml

---

QUESTION 6

Get list of all the pods showing name and namespace with a jsonpath expression.

Correct Answer: Check the answer in explanation.

kubectl get pods -o=jsonpath="{.items[*][\\'metadata.name\\' , \\'metadata.namespace\\']}"

QUESTION 7

List "nginx-dev" and "nginx-prod" pod and delete those pods

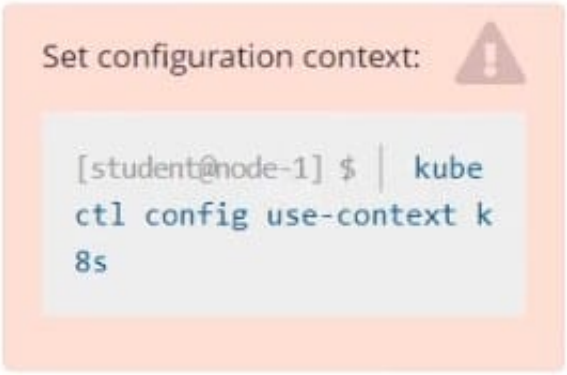
Correct Answer: Check the answer in explanation.

Solution

kubect1 get pods -o wide kubectl delete po "nginx-dev" kubectl delete po "nginx-prod"

QUESTION 8

CORRECT TEXT



Task Monitor the logs of pod bar and: Extract log lines corresponding to error file-not-found

Write them to /opt/KUTR00101/bar

Correct Answer: Check the answer in explanation.

kubectl logs bar | grep \\'unable-to-access-website\\' > /opt/KUTR00101/bar cat /opt/KUTR00101/bar

---

QUESTION 9

Check the Image version of nginx-dev pod using jsonpath

Correct Answer: Check the answer in explanation.

kubect1 get po nginx-dev -o jsonpath=\\'{.spec.containers[].image}{"\n"}\\'

QUESTION 10

SIMULATION

Create a deployment as follows:

1.

Name: nginx-app

2.

Using container nginx with version 1.11.10-alpine

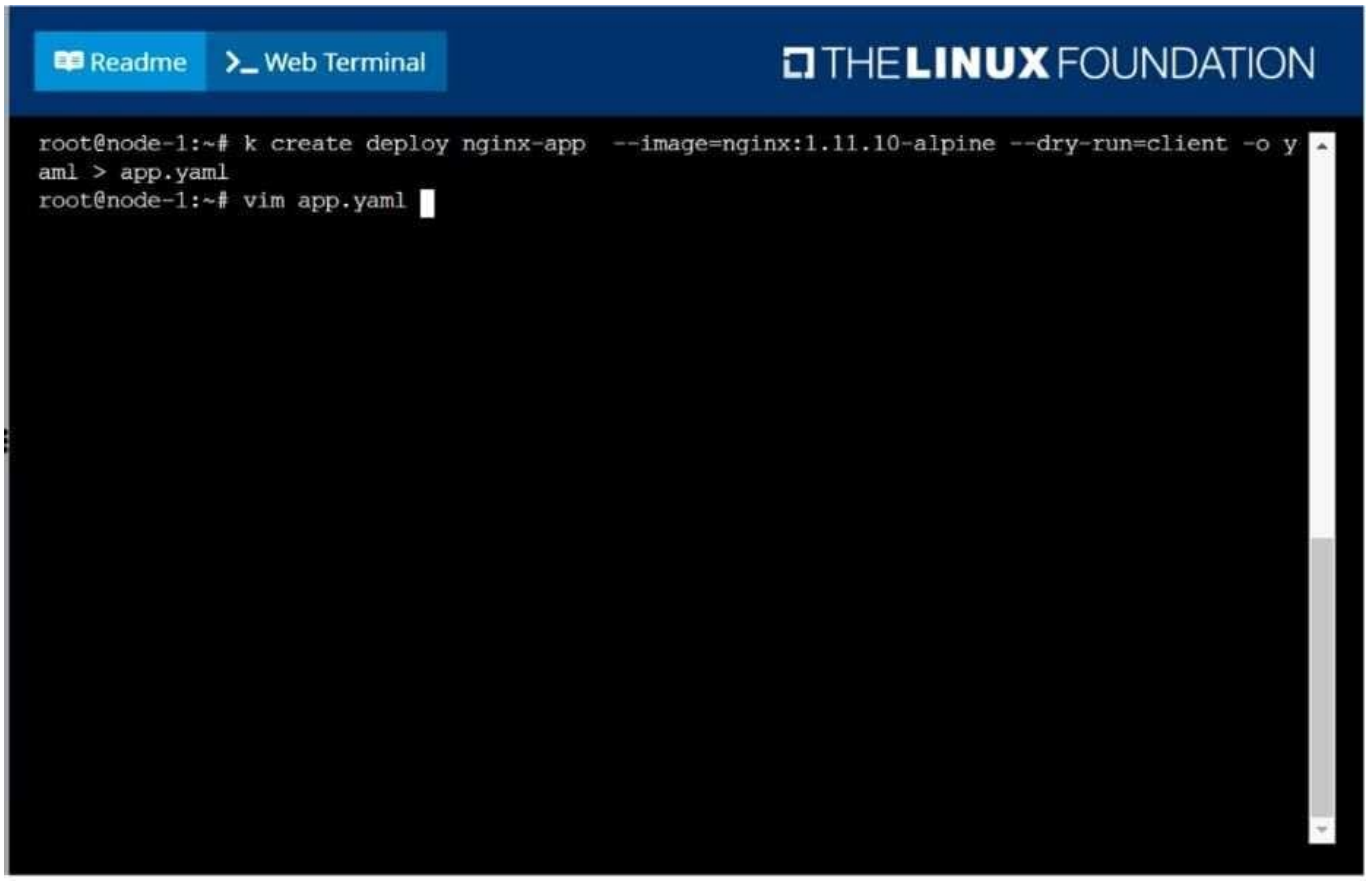
3.

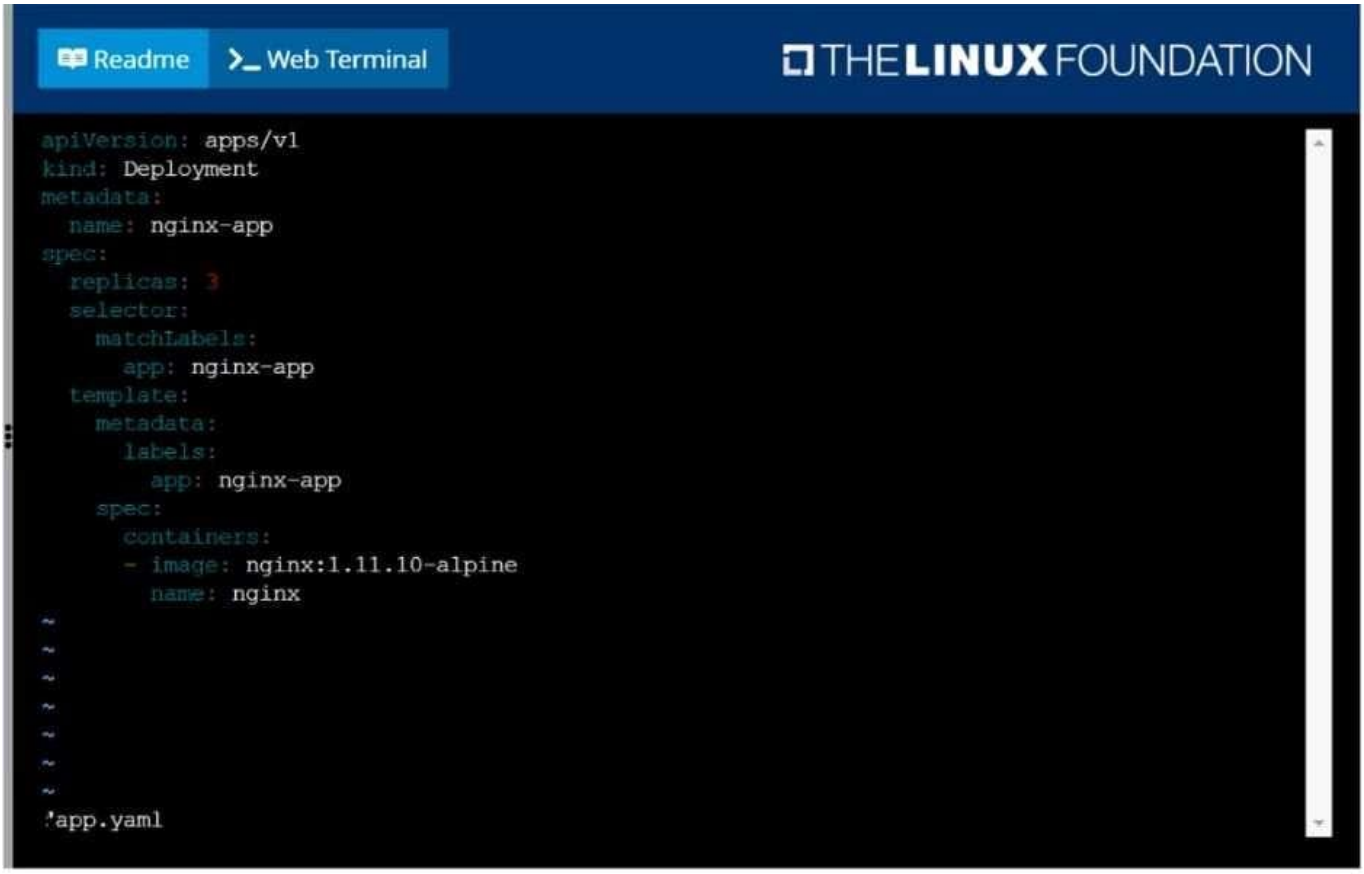
The deployment should contain 3 replicas

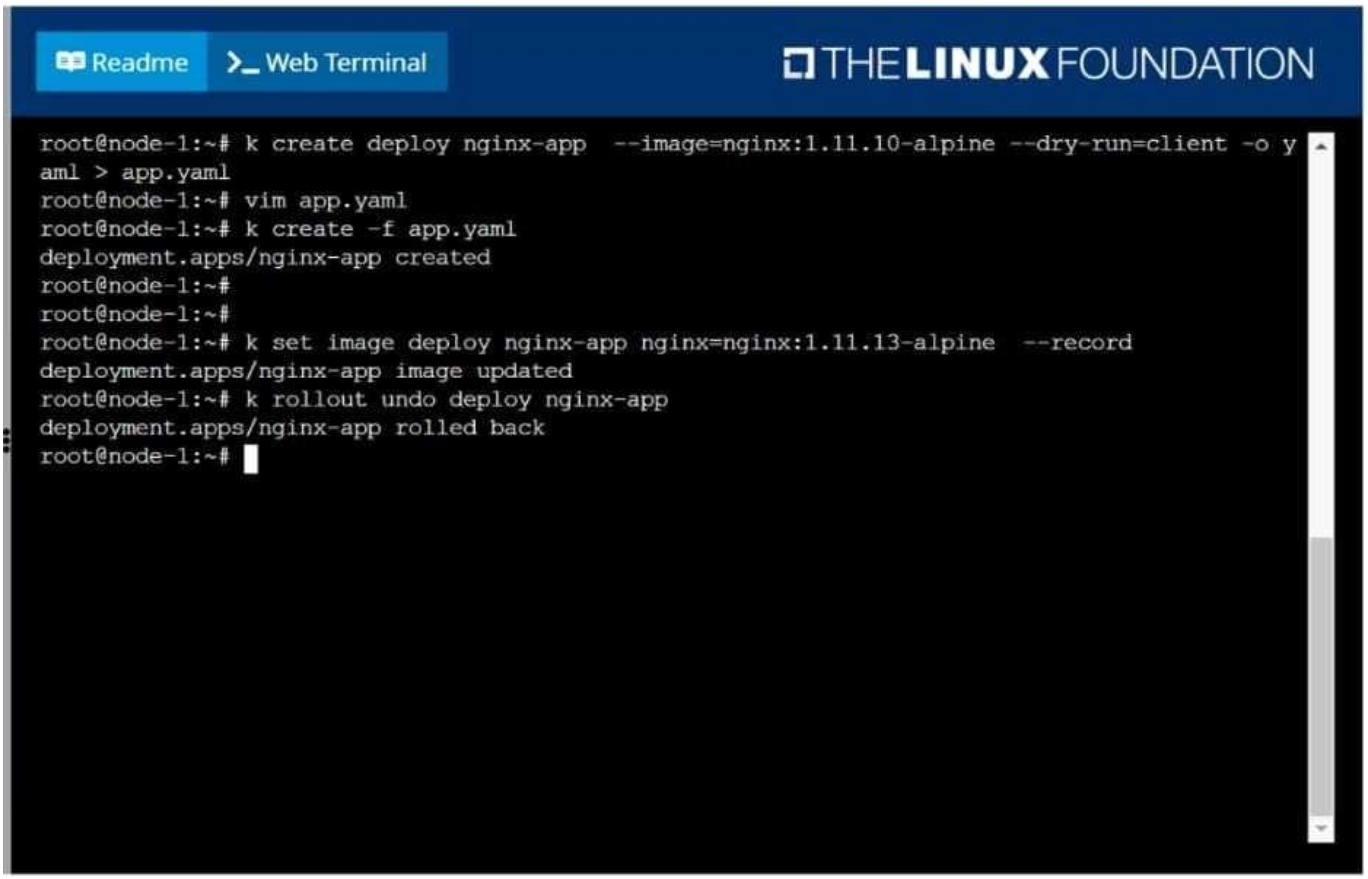
Next, deploy the application with new version 1.11.13-alpine, by performing a rolling update. Finally, rollback that update to the previous version 1.11.10-alpine.

Correct Answer: Check the answer in explanation.

Solution


Readme
Web Terminal
THE LINUX FOUNDATION
root@node-1:~# k create deploy nginx-app --image=nginx:1.11.10-alpine --dry-run=client -o y
aml > app.yaml
root@node-1:~# vim app.yaml


Readme
Web Terminal
THE LINUX FOUNDATION
apiVersion: apps/v1
kind: Deployment
metadata:
name: nginx-app
spec:
replicas: 3
selector:
matchLabels:
app: nginx-app
template:
metadata:
labels:
app: nginx-app
spec:
containers:
- image: nginx:1.11.10-alpine
name: nginx
~
~
~
~
~
~
~
'app.yaml

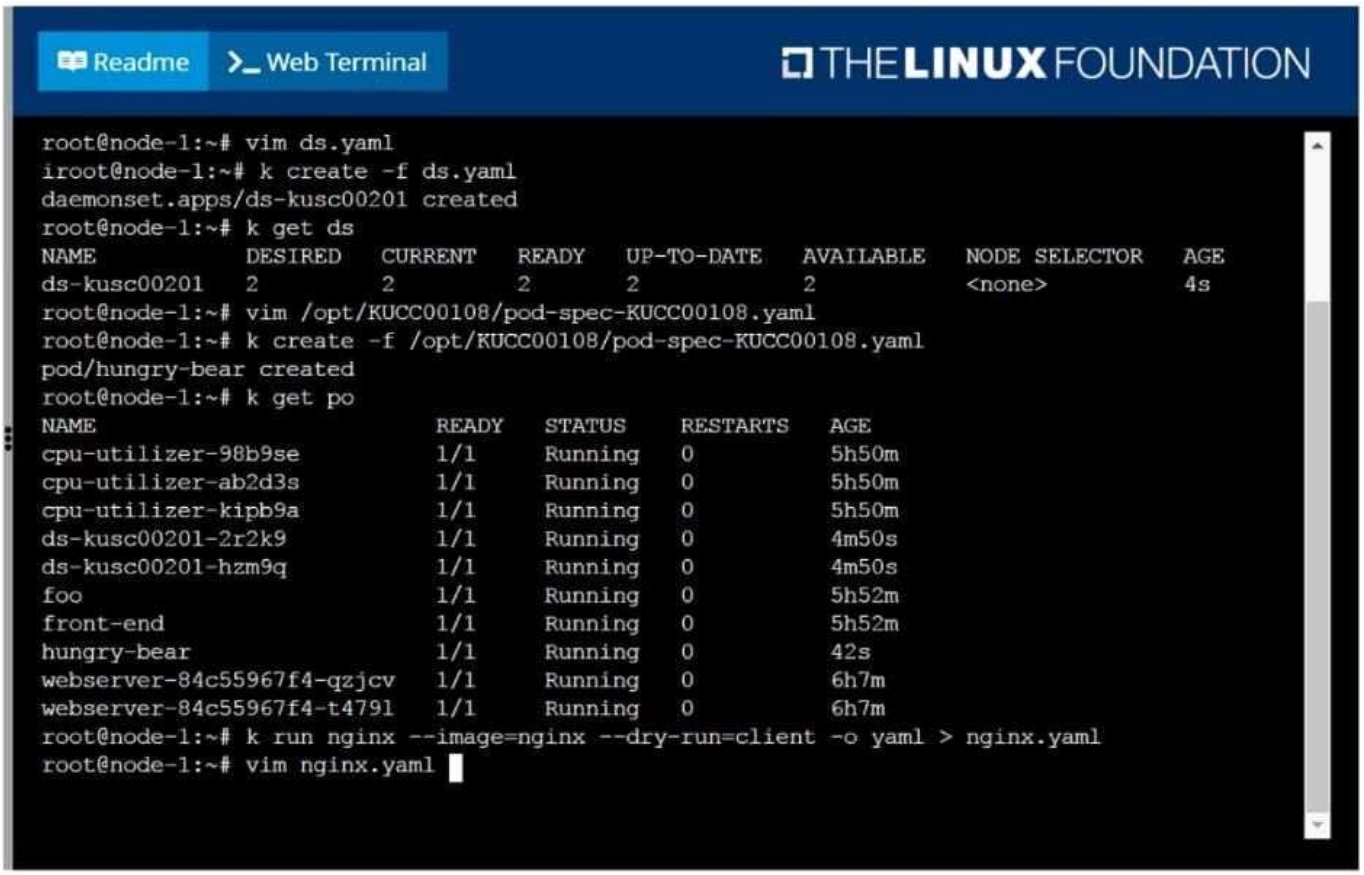

Readme
Web Terminal
THE LINUX FOUNDATION
root@node-1:~# k create deploy nginx-app --image=nginx:1.11.10-alpine --dry-run=client -o y
aml > app.yaml
root@node-1:~# vim app.yaml
root@node-1:~# k create -f app.yaml
deployment.apps/nginx-app created
root@node-1:~#
root@node-1:~#
root@node-1:~# k set image deploy nginx-app nginx=nginx:1.11.13-alpine --record
deployment.apps/nginx-app image updated
root@node-1:~# k rollout undo deploy nginx-app
deployment.apps/nginx-app rolled back
root@node-1:~#

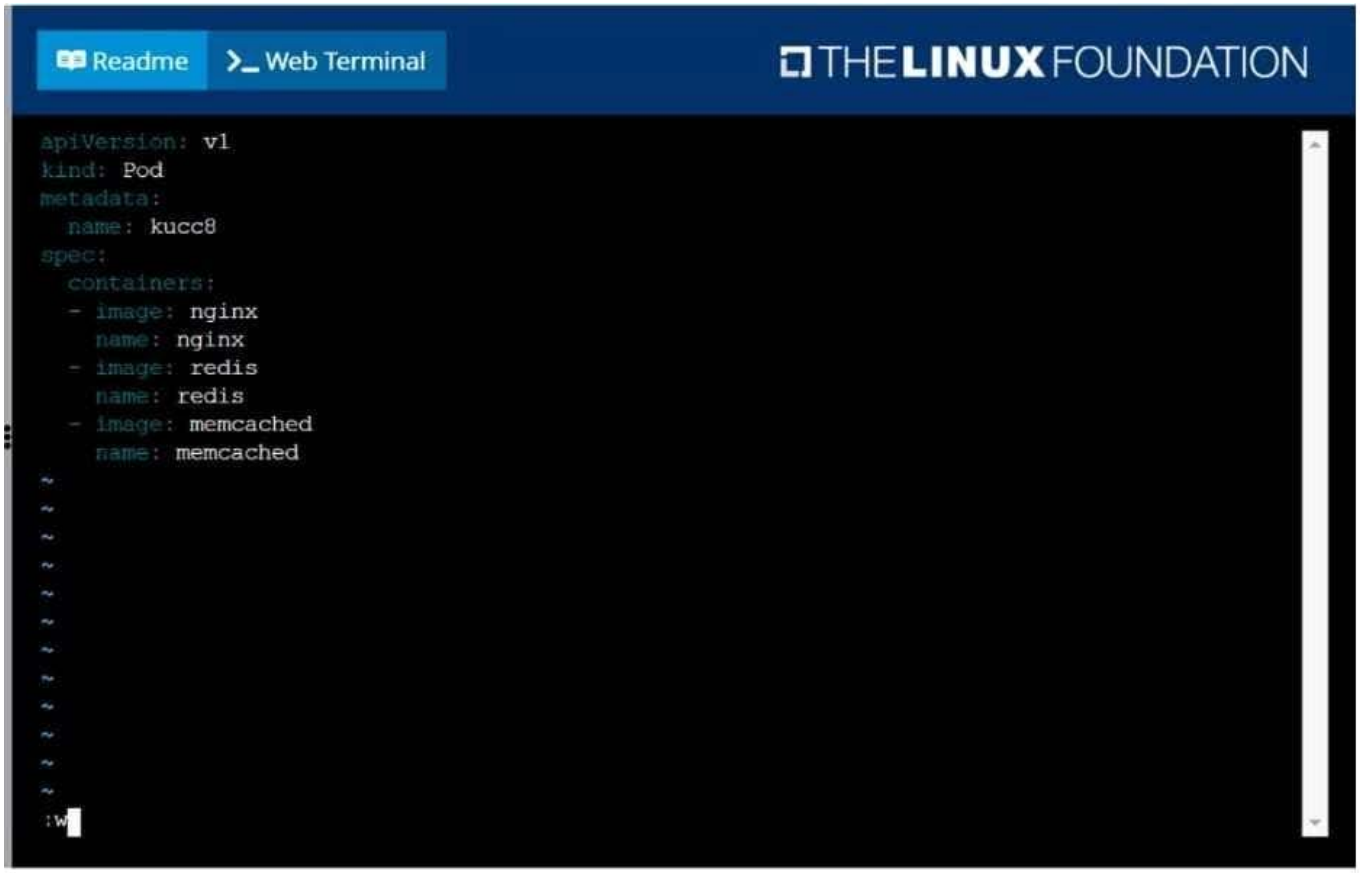
QUESTION 11

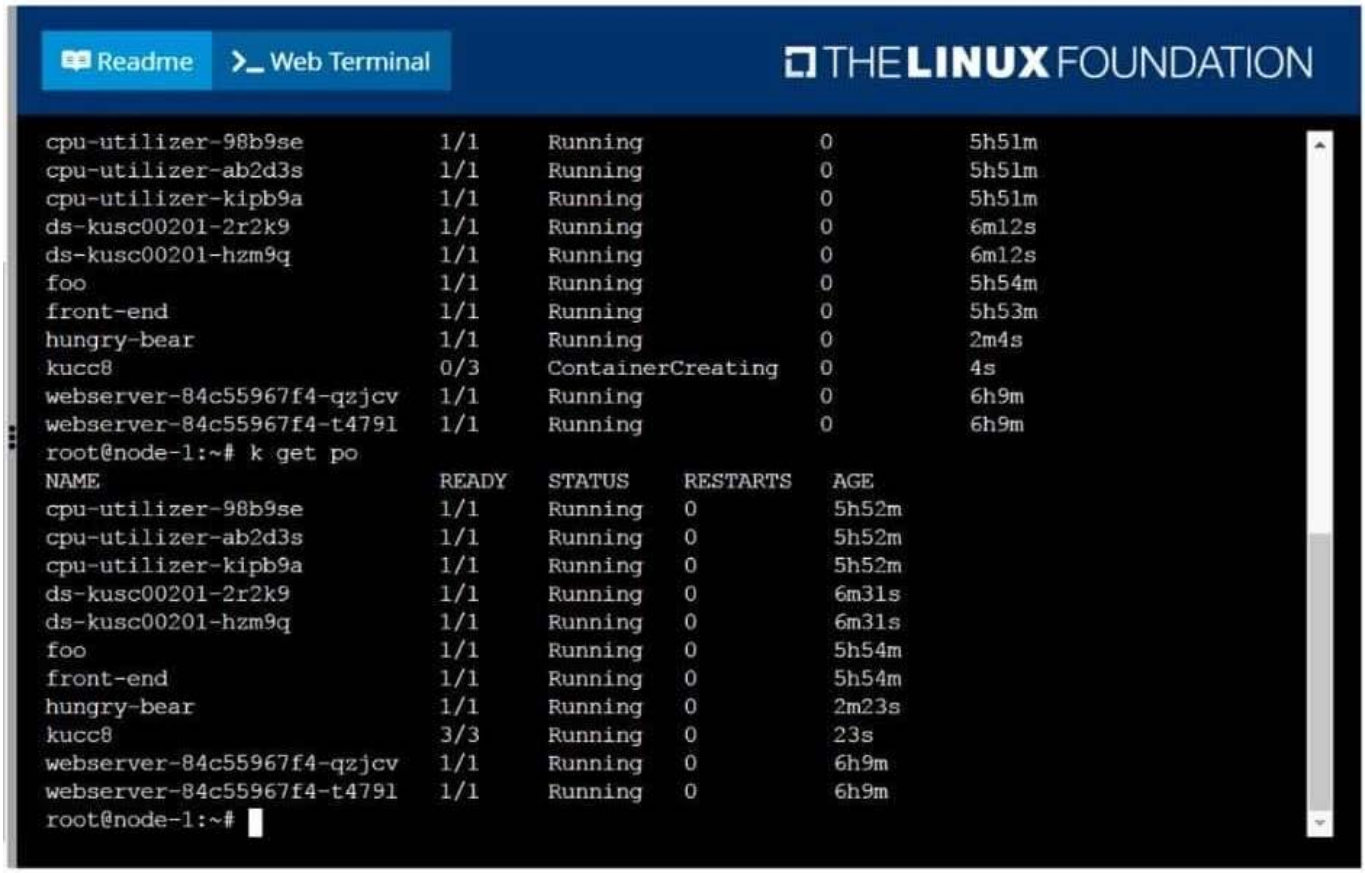
SIMULATION

Create a pod named kucc8 with a single app container for each of the following images running inside (there may be between 1 and 4 images specified): nginx + redis + memcached.

Correct Answer: Check the answer in explanation.




Readme
Web Terminal
THE LINUX FOUNDATION
apiVersion: v1
kind: Pod
metadata:
name: kucc8
spec:
containers:
- image: nginx
name: nginx
- image: redis
name: redis
- image: memcached
name: memcached
:w


Readme
Web Terminal
THE LINUX FOUNDATION
cpu-utilizer-98b9se 1/1 Running 0 5h51m
cpu-utilizer-ab2d3s 1/1 Running 0 5h51m
cpu-utilizer-kipb9a 1/1 Running 0 5h51m
ds-kusc00201-2r2k9 1/1 Running 0 6m12s
ds-kusc00201-hzm9q 1/1 Running 0 6m12s
foo 1/1 Running 0 5h54m
front-end 1/1 Running 0 5h53m
hungry-bear 1/1 Running 0 2m4s
kucc8 0/3 ContainerCreating 0 4s
webserver-84c55967f4-qzjcv 1/1 Running 0 6h9m
webserver-84c55967f4-t4791 1/1 Running 0 6h9m
root@node-1:~# k get po
NAME READY STATUS RESTARTS AGE
cpu-utilizer-98b9se 1/1 Running 0 5h52m
cpu-utilizer-ab2d3s 1/1 Running 0 5h52m
cpu-utilizer-kipb9a 1/1 Running 0 5h52m
ds-kusc00201-2r2k9 1/1 Running 0 6m31s
ds-kusc00201-hzm9q 1/1 Running 0 6m31s
foo 1/1 Running 0 5h54m
front-end 1/1 Running 0 5h54m
hungry-bear 1/1 Running 0 2m23s
kucc8 3/3 Running 0 23s
webserver-84c55967f4-qzjcv 1/1 Running 0 6h9m
webserver-84c55967f4-t4791 1/1 Running 0 6h9m
root@node-1:~#

QUESTION 12

SIMULATION

Create a Kubernetes secret as follows:

Name: super-secret

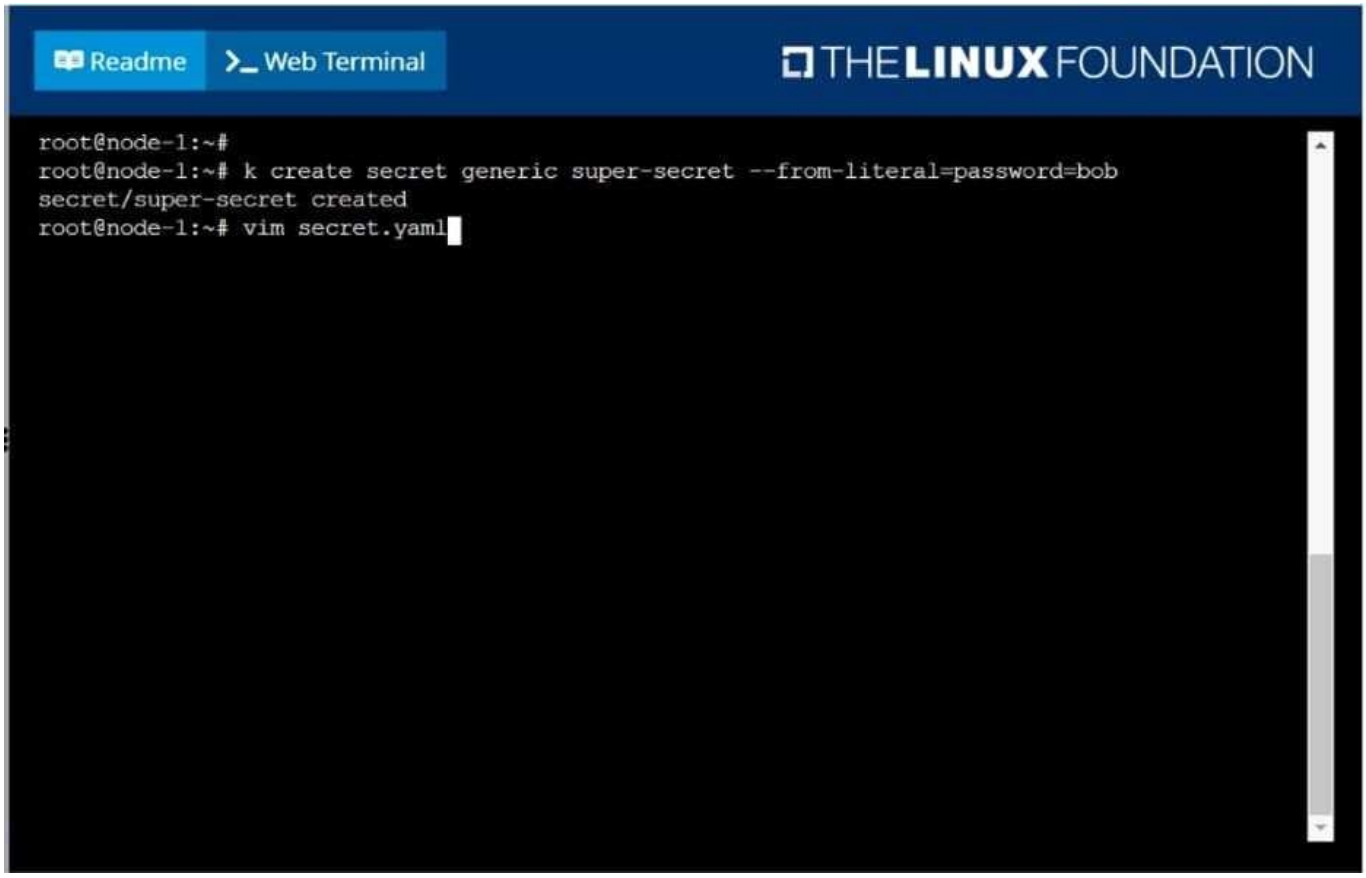
password: bob

Create a pod named pod-secrets-via-file, using the redis Image, which mounts a secret named super- secret at /secrets.

Create a second pod named pod-secrets-via-env, using the redis Image, which exports password as CONFIDENTIAL

Correct Answer: Check the answer in explanation.

Solution



Readme | Web Terminal — THE LINUX FOUNDATION

```
apiVersion: v1
kind: Pod
metadata:
  name: pod-secrets-via-file
spec:
  containers:
  - name: redis
    image: redis
    volumeMounts:
    - name: foo
      mountPath: "/secrets"
  volumes:
  - name: foo
    secret:
      secretName: super-secret
~
~
~
~
~
~
~
~
~
:w
```

Readme | Web Terminal — THE LINUX FOUNDATION

```
root@node-1:~# k create -f secret.yaml
pod/pod-secrets-via-file created
root@node-1:~# vim secret1.yaml
root@node-1:~# k create -f secret1.yaml
pod/pod-secrets-via-env created
root@node-1:~# k get po
NAME                          READY   STATUS    RESTARTS   AGE
cpu-utilizer-98b9se           1/1     Running   0          6h25m
cpu-utilizer-ab2d3s           1/1     Running   0          6h25m
cpu-utilizer-kipb9a           1/1     Running   0          6h25m
ds-kusc00201-2r2k9            1/1     Running   0          40m
ds-kusc00201-hzm9q            1/1     Running   0          40m
foo                           1/1     Running   0          6h28m
front-end                     1/1     Running   0          6h27m
hungry-bear                   1/1     Running   0          36m
kucc8                         3/3     Running   0          34m
nginx-app-848cfcf495-9prjh    1/1     Running   0          19m
nginx-app-848cfcf495-gl2kh    1/1     Running   0          19m
nginx-app-848cfcf495-pg2c8    1/1     Running   0          19m
nginx-kusc00101               1/1     Running   0          26m
pod-secrets-via-env           1/1     Running   0          4s
pod-secrets-via-file          1/1     Running   0          106s
webserver-84c55967f4-qzjcv    1/1     Running   0          6h43m
webserver-84c55967f4-t4791    1/1     Running   0          6h43m
root@node-1:~#
```

QUESTION 13

CORRECT TEXT

List "nginx-dev" and "nginx-prod" pod and delete those pods

Correct Answer: Check the answer in explanation.

kubect1 get pods -o wide

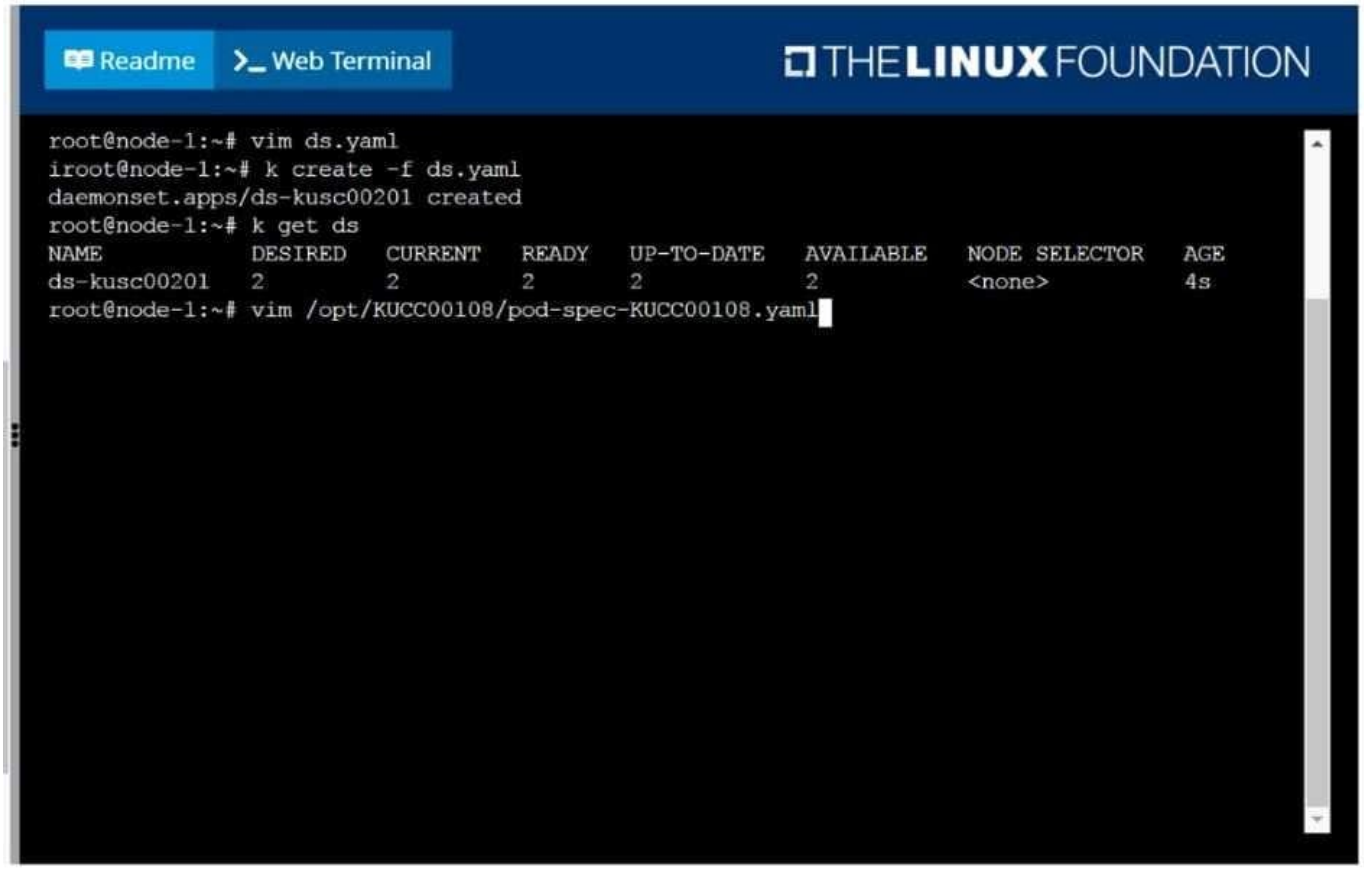
kubectl delete po "nginx-dev"kubectl delete po "nginx-prod"

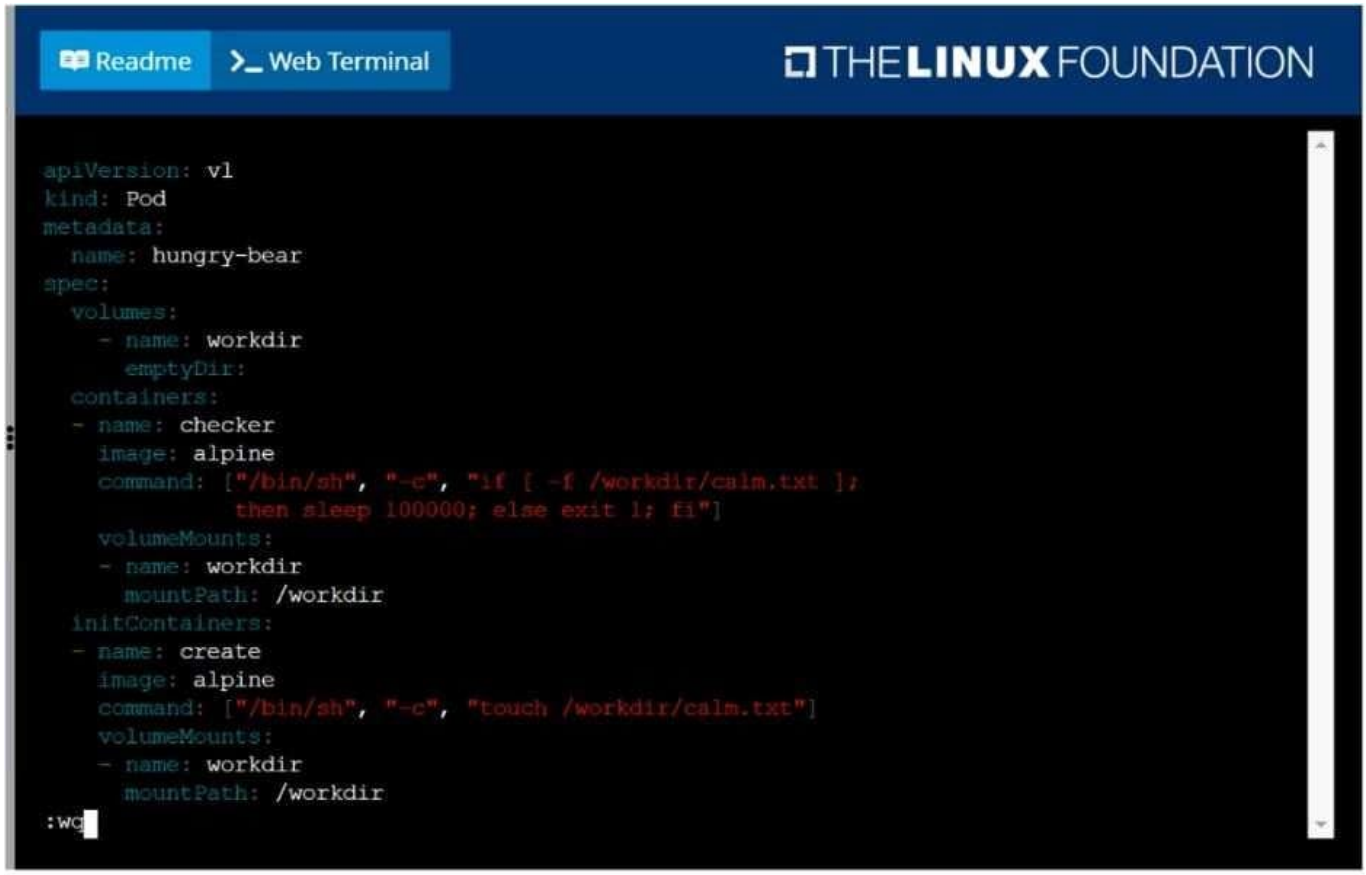
QUESTION 14

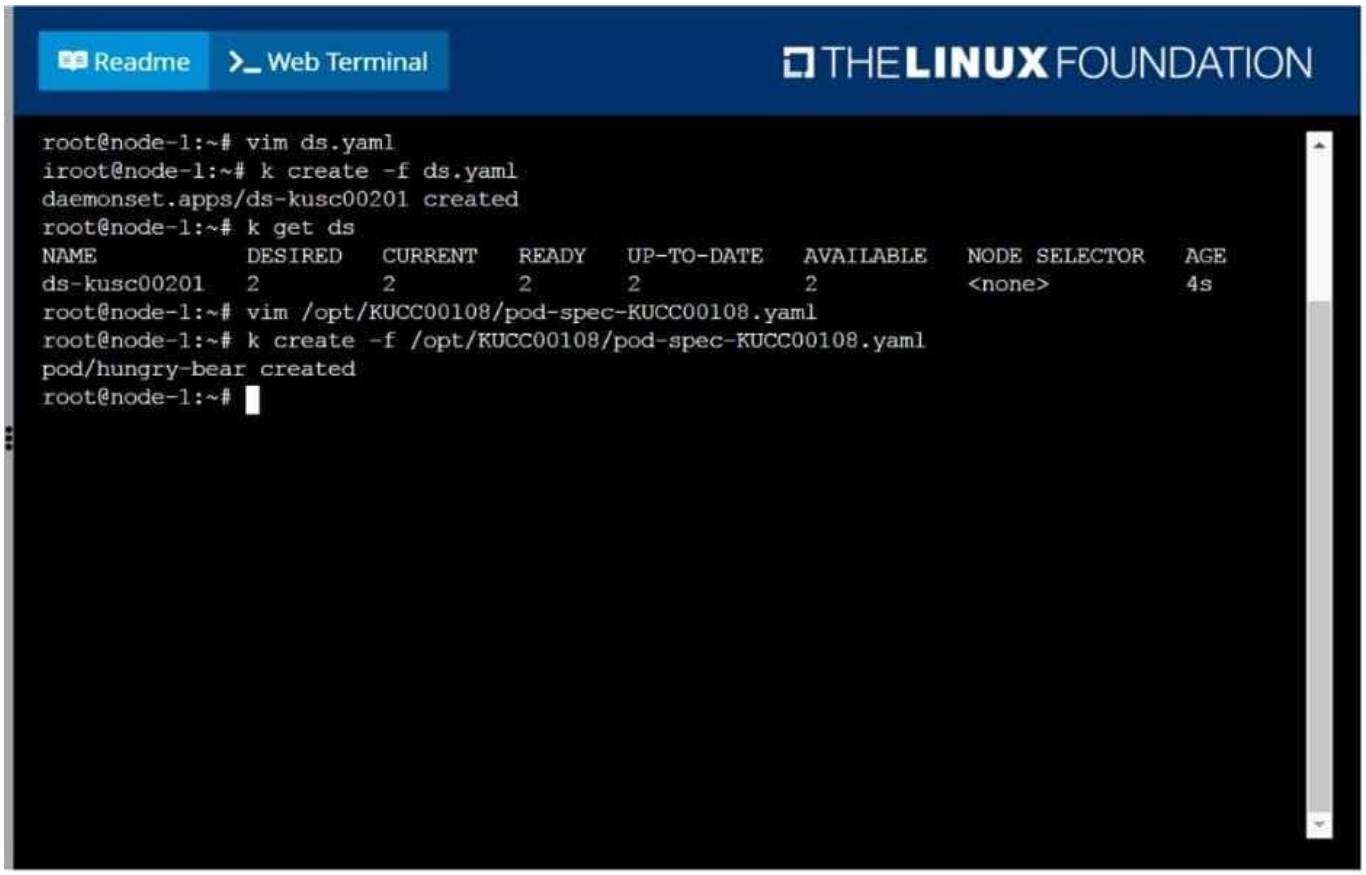
SIMULATION

Perform the following tasks: Add an init container to hungry-bear (which has been defined in spec file /opt/KUCC00108/pod- spec-KUCC00108.yaml) The init container should create an empty file named /workdir/calm.txt If /workdir/calm.txt is not detected, the pod should exit Once the spec file has been updated with the init container definition, the pod should be created

Correct Answer: Check the answer in explanation.




Readme
Web Terminal
THE LINUX FOUNDATION
apiVersion: v1
kind: Pod
metadata:
name: hungry-bear
spec:
volumes:
- name: workdir
emptyDir:
containers:
- name: checker
image: alpine
command: ["/bin/sh", "-c", "if [-f /workdir/calm.txt];
then sleep 100000; else exit 1; fi"]
volumeMounts:
- name: workdir
mountPath: /workdir
initContainers:
- name: create
image: alpine
command: ["/bin/sh", "-c", "touch /workdir/calm.txt"]
volumeMounts:
- name: workdir
mountPath: /workdir
:wq


Readme
Web Terminal
THE LINUX FOUNDATION
root@node-1:~# vim ds.yaml
iroot@node-1:~# k create -f ds.yaml
daemonset.apps/ds-kusc00201 created
root@node-1:~# k get ds
NAME DESIRED CURRENT READY UP-TO-DATE AVAILABLE NODE SELECTOR AGE
ds-kusc00201 2 2 2 2 2 <none> 4s
root@node-1:~# vim /opt/KUCC00108/pod-spec-KUCC00108.yaml
root@node-1:~# k create -f /opt/KUCC00108/pod-spec-KUCC00108.yaml
pod/hungry-bear created
root@node-1:~#

QUESTION 15

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"