

# Linux-Foundation

## Exam Questions CKA

Certified Kubernetes Administrator (CKA) Program



**NEW QUESTION 1**

CORRECT TEXT

Create a pod with image nginx called nginx and allow traffic on port 80

- A. Mastered
- B. Not Mastered

**Answer:** A

**Explanation:**

kubectrl run nginx --image=nginx --restart=Never --port=80

**NEW QUESTION 2**

CORRECT TEXT

Create a deployment as follows:

? Name: nginx-random

? Exposed via a service nginx-random

? Ensure that the service & pod are accessible via their respective DNS records

? The container(s) within any pod(s) running as a part of this deployment should use the nginx Image

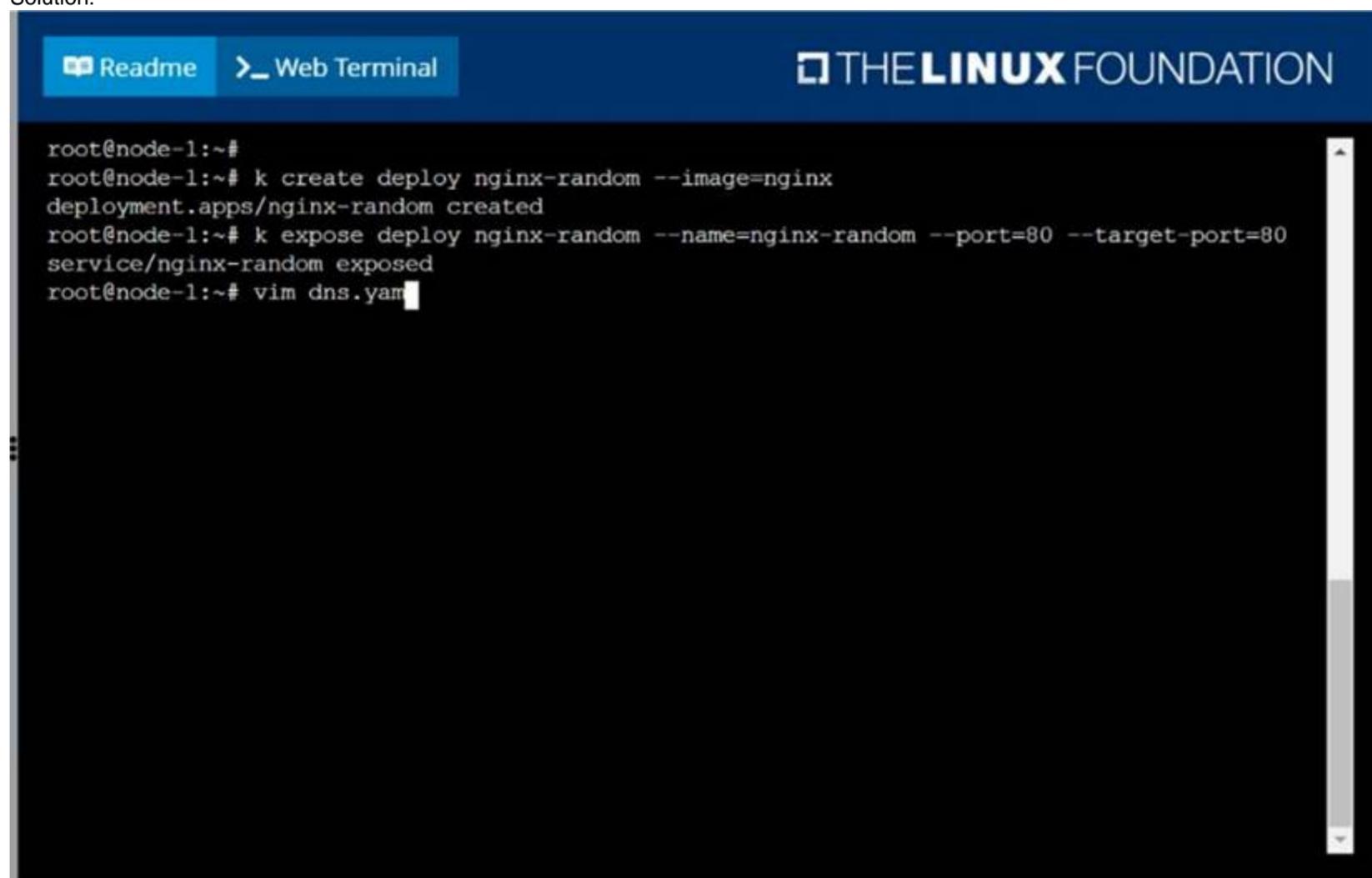
Next, use the utility nslookup to look up the DNS records of the service & pod and write the output to /opt/KUNW00601/service.dns and /opt/KUNW00601/pod.dns respectively.

- A. Mastered
- B. Not Mastered

**Answer:** A

**Explanation:**

Solution:



The screenshot shows a web terminal interface with a blue header containing 'Readme' and 'Web Terminal' buttons, and 'THE LINUX FOUNDATION' logo. The terminal output is as follows:

```

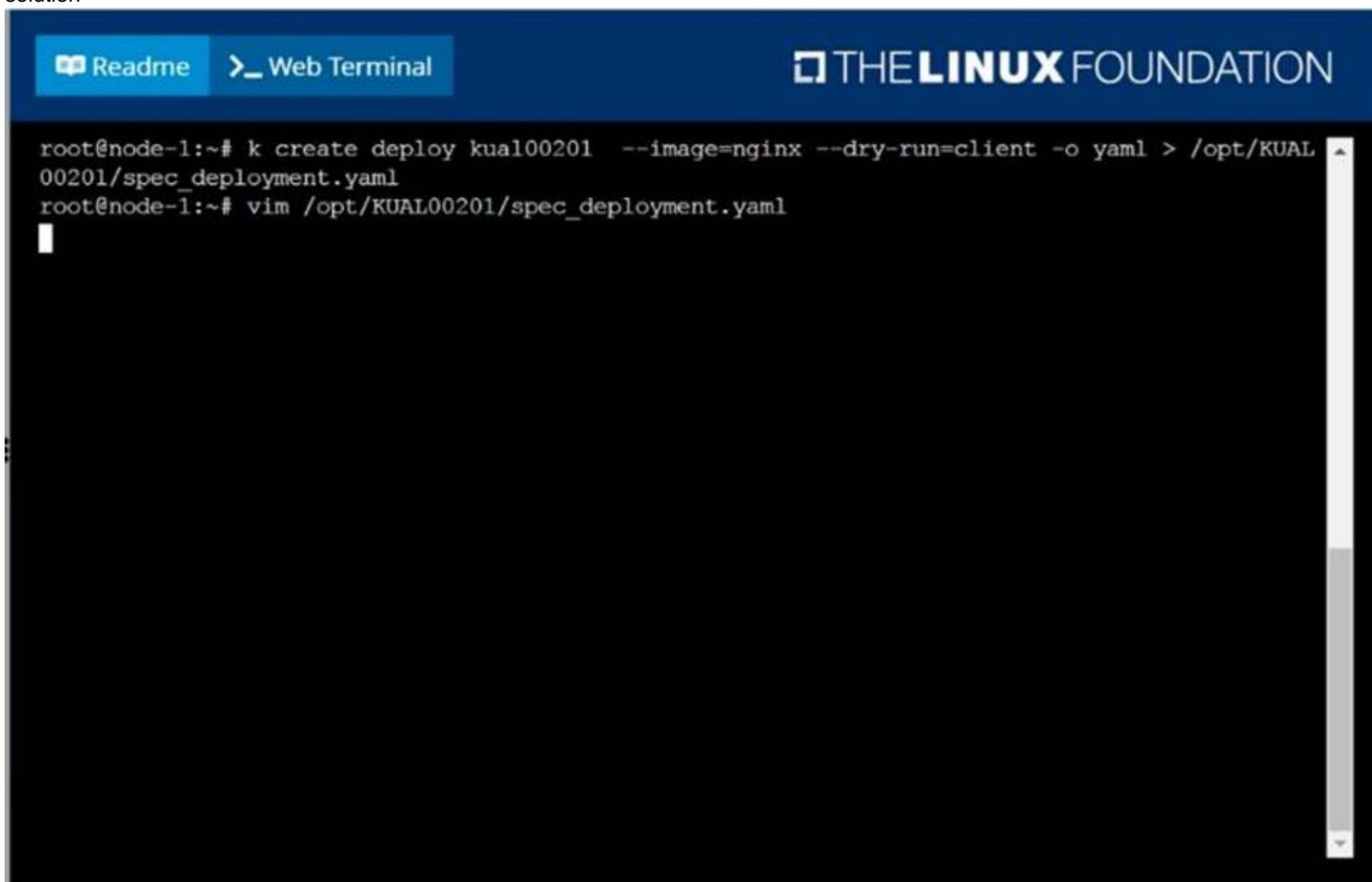
root@node-1:~#
root@node-1:~# k create deploy nginx-random --image=nginx
deployment.apps/nginx-random created
root@node-1:~# k expose deploy nginx-random --name=nginx-random --port=80 --target-port=80
service/nginx-random exposed
root@node-1:~# vim dns.yaml

```

F:\Work\Data Entry Work\Data Entry\20200827\CKA\17 C.JPG



**Explanation:**  
 solution

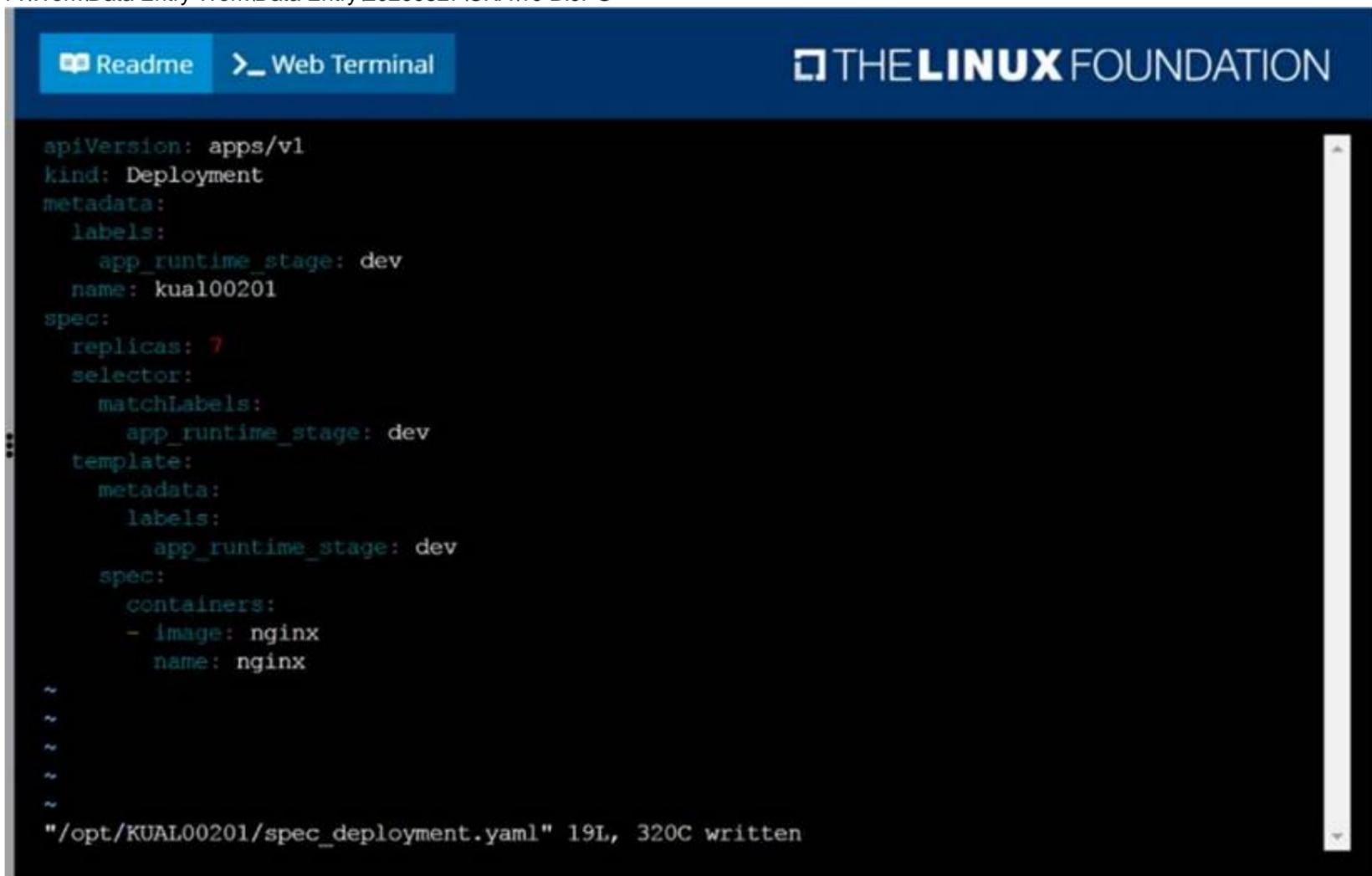


The screenshot shows a web terminal interface with a blue header containing 'THE LINUX FOUNDATION' logo and navigation buttons for 'Readme' and 'Web Terminal'. The terminal output shows the following commands and their results:

```

root@node-1:~# k create deploy kual00201 --image=nginx --dry-run=client -o yaml > /opt/KUAL
00201/spec_deployment.yaml
root@node-1:~# vim /opt/KUAL00201/spec_deployment.yaml
  
```

F:\Work\Data Entry Work\Data Entry\20200827\CKA\10 B.JPG



The screenshot shows the same web terminal interface as above, but now displaying the contents of the deployment file created in the previous step:

```

apiVersion: apps/v1
kind: Deployment
metadata:
  labels:
    app_runtime_stage: dev
  name: kual00201
spec:
  replicas: 7
  selector:
    matchLabels:
      app_runtime_stage: dev
  template:
    metadata:
      labels:
        app_runtime_stage: dev
    spec:
      containers:
      - image: nginx
        name: nginx
~
~
~
~
~
"/opt/KUAL00201/spec_deployment.yaml" 19L, 320C written
  
```

F:\Work\Data Entry Work\Data Entry\20200827\CKA\10 C.JPG

**NEW QUESTION 4**

CORRECT TEXT

Scale the deployment webserver to 6 pods.

- A. Mastered
- B. Not Mastered

**Answer:** A

**Explanation:**  
 solution

```

Readme Web Terminal THE LINUX FOUNDATION

root@node-1:~# k scale deploy webserver --replicas=6
deployment.apps/webserver scaled
root@node-1:~# k get deploy
NAME          READY   UP-TO-DATE   AVAILABLE   AGE
nginx-app     3/3     3             3           29m
webserver     6/6     6             6           6h50m
root@node-1:~#

```

F:\Work\Data Entry Work\Data Entry\20200827\CKA\14 B.JPG

**NEW QUESTION 5**

CORRECT TEXT

Score: 7%



Task

Create a new NetworkPolicy named allow-port-from-namespace in the existing namespace echo. Ensure that the new NetworkPolicy allows Pods in namespace my-app to connect to port 9000 of Pods in namespace echo.

Further ensure that the new NetworkPolicy:

- does not allow access to Pods, which don't listen on port 9000
- does not allow access from Pods, which are not in namespace my-app

- A. Mastered
- B. Not Mastered

**Answer: A**

**Explanation:**

Solution:

```

#network.yaml
apiVersion: networking.k8s.io/v1
kind: NetworkPolicy
metadata:
  name: allow-port-from-namespace
  namespace: internal
spec:
  podSelector:
    matchLabels: {
    }
  policyTypes:

```

```
- Ingress
ingress:
- from:
- podSelector: {
}
ports:
- protocol: TCP
port: 8080
#spec.podSelector namespace pod
kubectl create -f network.yaml
```

**NEW QUESTION 6**

CORRECT TEXT

Score: 7%



**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.

- A. Mastered
- B. Not Mastered

**Answer:** A

**Explanation:**

**Solution:**  
 kubectl get deploy front-end  
 kubectl edit deploy front-end -o yaml  
 #port specification named http  
 #service.yaml  
 apiVersion: v1  
 kind: Service  
 metadata:  
 name: front-end-svc  
 labels:  
 app: nginx  
 spec:  
 ports:  
 - port: 80  
 protocol: tcp  
 name: http  
 selector:  
 app: nginx  
 type: NodePort  
 # kubectl create -f service.yaml  
 # kubectl get svc  
 # port specification named http  
 kubectl expose deployment front-end --name=front-end-svc --port=80 --target-port=80 --type=NodePort

**NEW QUESTION 7**

CORRECT TEXT

Create an nginx pod and list the pod with different levels of verbosity

- A. Mastered
- B. Not Mastered

**Answer:** A

**Explanation:**

// create a pod  
 kubectl run nginx --image=nginx --restart=Never --port=80  
 // List the pod with different verbosity

```
kubectl get po nginx --v=7
kubectl get po nginx --v=8
kubectl get po nginx --v=9
```

**NEW QUESTION 8**

CORRECT TEXT

Create a file:

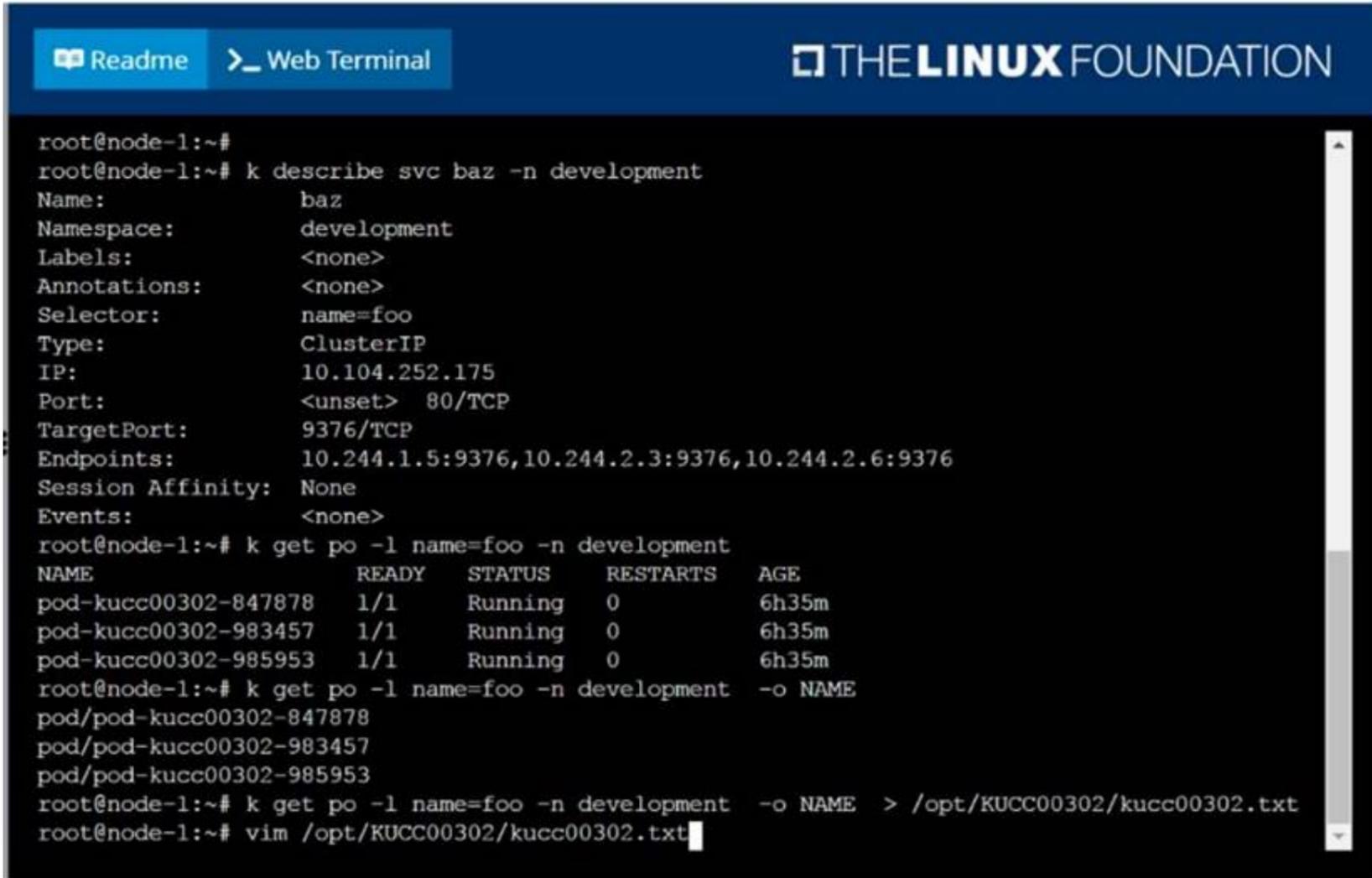
/opt/KUCC00302/kucc00302.txt that lists all pods that implement service baz in namespace development.  
 The format of the file should be one pod name per line.

- A. Mastered
- B. Not Mastered

Answer: A

**Explanation:**

solution



```

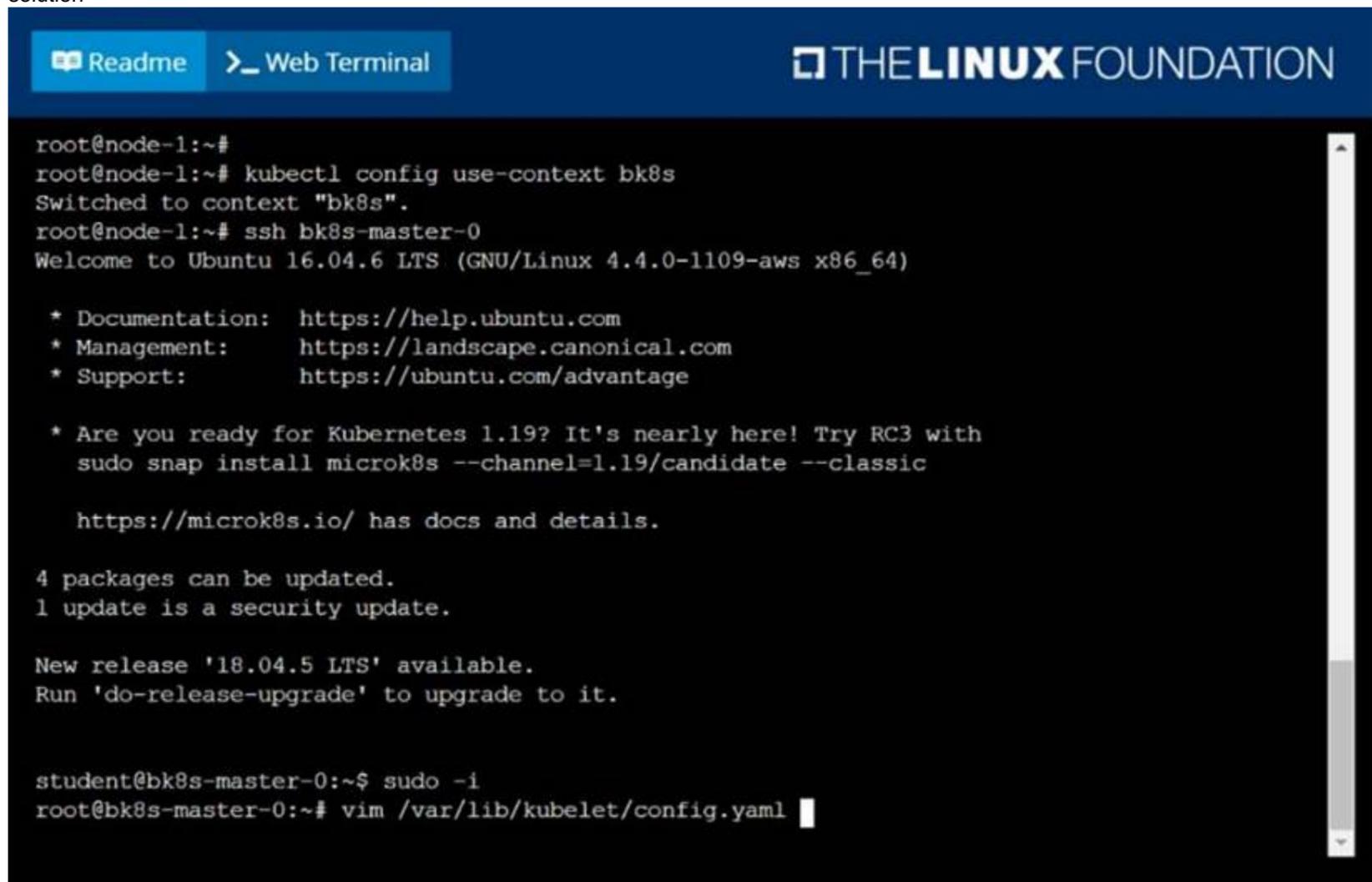
root@node-1:~#
root@node-1:~# k describe svc baz -n development
Name:          baz
Namespace:     development
Labels:        <none>
Annotations:   <none>
Selector:      name=foo
Type:          ClusterIP
IP:            10.104.252.175
Port:          <unset> 80/TCP
TargetPort:    9376/TCP
Endpoints:     10.244.1.5:9376,10.244.2.3:9376,10.244.2.6:9376
Session Affinity: None
Events:        <none>
root@node-1:~# k get po -l name=foo -n development
NAME                    READY   STATUS    RESTARTS   AGE
pod-kucc00302-847878    1/1     Running   0           6h35m
pod-kucc00302-983457    1/1     Running   0           6h35m
pod-kucc00302-985953    1/1     Running   0           6h35m
root@node-1:~# k get po -l name=foo -n development -o NAME
pod/pod-kucc00302-847878
pod/pod-kucc00302-983457
pod/pod-kucc00302-985953
root@node-1:~# k get po -l name=foo -n development -o NAME > /opt/KUCC00302/kucc00302.txt
root@node-1:~# vim /opt/KUCC00302/kucc00302.txt
    
```

F:\Work\Data Entry Work\Data Entry\20200827\CKA\11 B.JPG





Explanation:  
 solution



The screenshot shows a web terminal interface with a blue header containing 'THE LINUX FOUNDATION' logo and navigation buttons for 'Readme' and 'Web Terminal'. The terminal output shows the following sequence of commands and responses:

```

root@node-1:~#
root@node-1:~# kubectl config use-context bk8s
Switched to context "bk8s".
root@node-1:~# ssh bk8s-master-0
Welcome to Ubuntu 16.04.6 LTS (GNU/Linux 4.4.0-1109-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

 * Are you ready for Kubernetes 1.19? It's nearly here! Try RC3 with
   sudo snap install microk8s --channel=1.19/candidate --classic

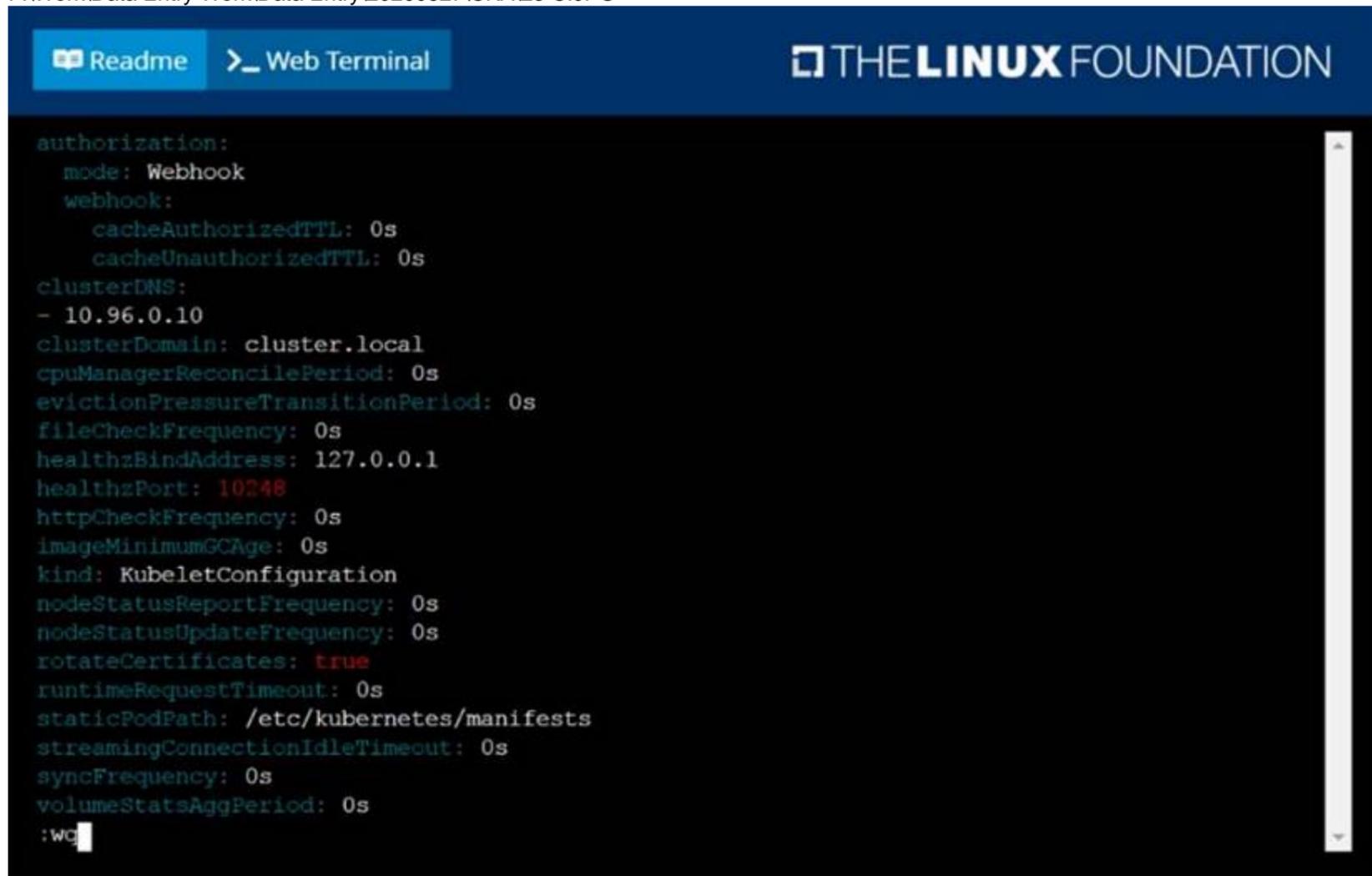
   https://microk8s.io/ has docs and details.

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

New release '18.04.5 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

student@bk8s-master-0:~$ sudo -i
root@bk8s-master-0:~# vim /var/lib/kubelet/config.yaml
  
```

F:\Work\Data Entry Work\Data Entry\20200827\CKA\23 C.JPG



The screenshot shows a web terminal interface with a blue header containing 'THE LINUX FOUNDATION' logo and navigation buttons for 'Readme' and 'Web Terminal'. The terminal output shows the contents of the kubelet configuration file:

```

authorization:
  mode: Webhook
  webhook:
    cacheAuthorizedTTL: 0s
    cacheUnauthorizedTTL: 0s
clusterDNS:
- 10.96.0.10
clusterDomain: cluster.local
cpuManagerReconcilePeriod: 0s
evictionPressureTransitionPeriod: 0s
fileCheckFrequency: 0s
healthzBindAddress: 127.0.0.1
healthzPort: 10248
httpCheckFrequency: 0s
imageMinimumGCAge: 0s
kind: KubeletConfiguration
nodeStatusReportFrequency: 0s
nodeStatusUpdateFrequency: 0s
rotateCertificates: true
runtimeRequestTimeout: 0s
staticPodPath: /etc/kubernetes/manifests
streamingConnectionIdleTimeout: 0s
syncFrequency: 0s
volumeStatsAggPeriod: 0s
:wg
  
```

F:\Work\Data Entry Work\Data Entry\20200827\CKA\23 D.JPG

```

Readme Web Terminal THE LINUX FOUNDATION

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

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

New release '18.04.5 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

student@bk8s-master-0:~$ sudo -i
root@bk8s-master-0:~# vim /var/lib/kubelet/config.yaml
root@bk8s-master-0:~# systemctl restart kubelet
root@bk8s-master-0:~# systemctl enable kubelet
root@bk8s-master-0:~# kubectl get nodes

NAME             STATUS    ROLES    AGE   VERSION
bk8s-master-0   Ready    master   77d   v1.18.2
bk8s-node-0     Ready    <none>   77d   v1.18.2
root@bk8s-master-0:~#
root@bk8s-master-0:~# exit
logout
student@bk8s-master-0:~$ exit
logout
Connection to 10.250.4.77 closed.
root@node-1:~#

```

F:\Work\Data Entry Work\Data Entry\20200827\CKA\23 E.JPG

**NEW QUESTION 10**

CORRECT TEXT

Create a pod as follows:

? Name: mongo

? Using Image: mongo

? In a new Kubernetes namespace named: my-website

- A. Mastered
- B. Not Mastered

**Answer: A**

**Explanation:**

solution

```

Readme Web Terminal THE LINUX FOUNDATION

root@node-1:~#
root@node-1:~#
root@node-1:~# k create ns my-website
namespace/my-website created
root@node-1:~# k run mongo --image=mongo -n my-website
pod/mongo created
root@node-1:~# k get po -n my-website
NAME     READY   STATUS              RESTARTS   AGE
mongo    0/1     ContainerCreating   0           4s
root@node-1:~#

```

F:\Work\Data Entry Work\Data Entry\20200827\CKA\9 B.JPG

**NEW QUESTION 13**

CORRECT TEXT

Create a deployment as follows:

? Name: nginx-app

? Using container nginx with version 1.11.10-alpine

? The deployment should contain 3 replicas

Next, deploy the application with new version 1.11.13-alpine, by performing a rolling update.

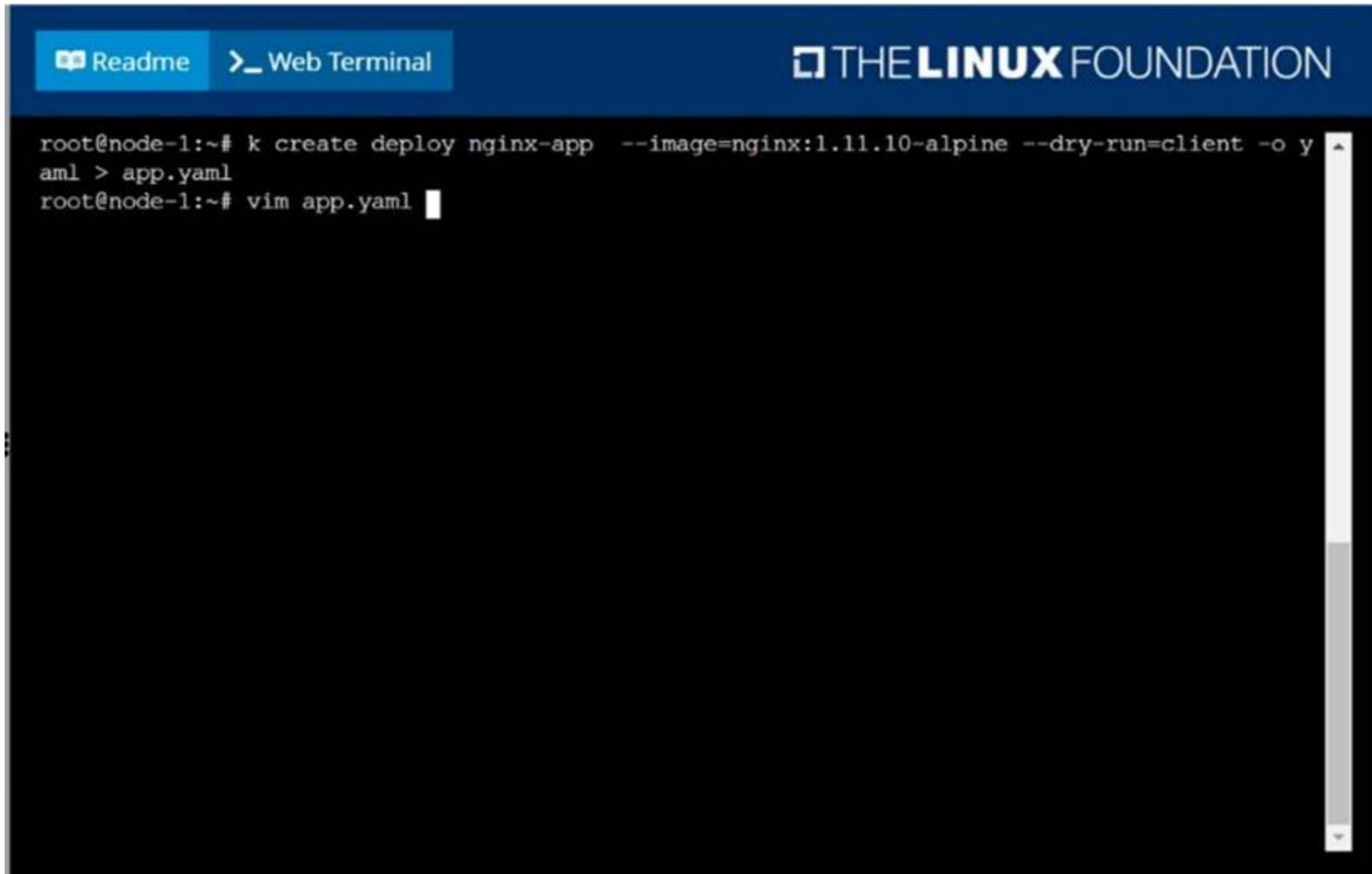
Finally, rollback that update to the previous version 1.11.10-alpine.

- A. Mastered
- B. Not Mastered

**Answer:** A

**Explanation:**

solution



The screenshot shows a web terminal window with a dark background and white text. At the top, there is a blue header bar with the text 'THE LINUX FOUNDATION' on the right and 'Readme' and 'Web Terminal' on the left. The terminal content shows the following commands and output:

```

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
  
```

F:\Work\Data Entry Work\Data Entry\20200827\CKA\7 B.JPG



**NEW QUESTION 19**

CORRECT TEXT

Create a pod as follows:

? Name: non-persistent-redis

? container Image: redis

? Volume with name: cache-control

? Mount path: /data/redis

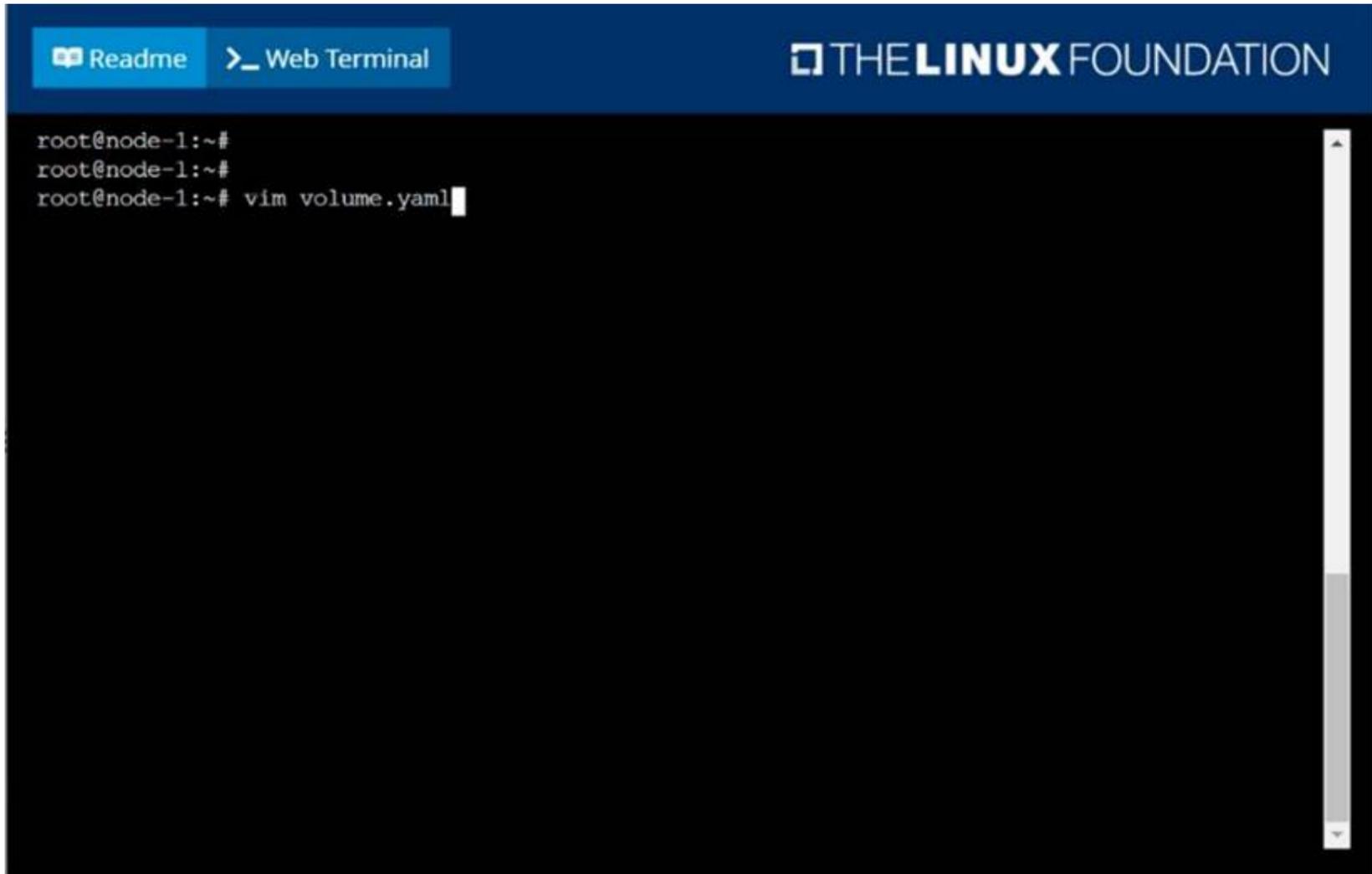
The pod should launch in the staging namespace and the volume must not be persistent.

- A. Mastered
- B. Not Mastered

**Answer:** A

**Explanation:**

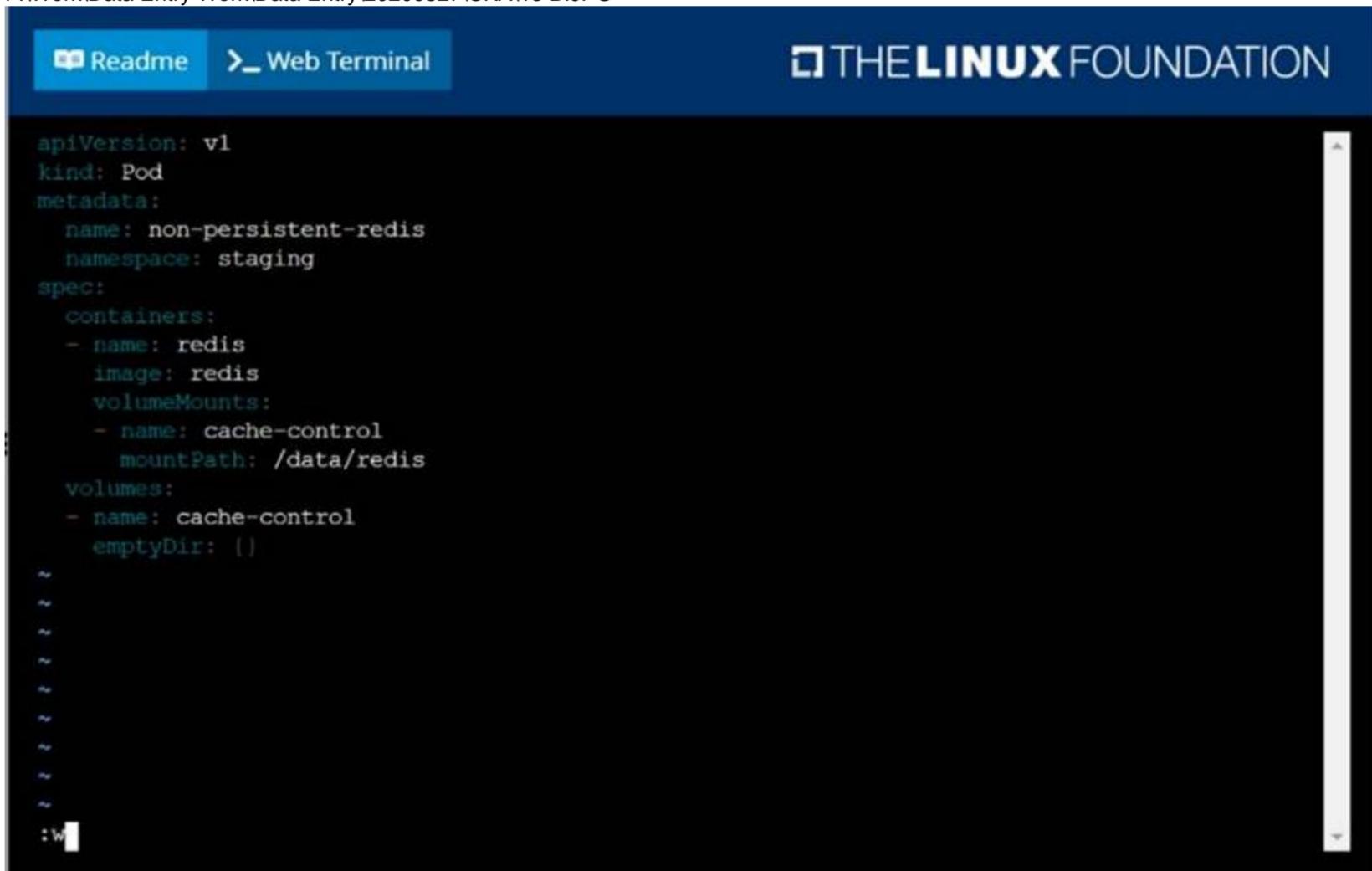
solution



The screenshot shows a web terminal interface with a blue header containing 'Readme' and 'Web Terminal' buttons, and 'THE LINUX FOUNDATION' logo. The terminal content is as follows:

```
root@node-1:~#
root@node-1:~#
root@node-1:~# vim volume.yaml
```

F:\Work\Data Entry Work\Data Entry\20200827\CKA\13 B.JPG



The screenshot shows the same web terminal interface as above, but now displaying the contents of the 'volume.yaml' file:

```
apiVersion: v1
kind: Pod
metadata:
  name: non-persistent-redis
  namespace: staging
spec:
  containers:
  - name: redis
    image: redis
    volumeMounts:
    - name: cache-control
      mountPath: /data/redis
  volumes:
  - name: cache-control
    emptyDir: {}
~
~
~
~
~
~
~
~
~
~
:w
```

F:\Work\Data Entry Work\Data Entry\20200827\CKA\13 C.JPG

[Readme](#) > [Web Terminal](#)
THE LINUX FOUNDATION

```

root@node-1:~#
root@node-1:~#
root@node-1:~# vim volume.yaml
root@node-1:~# k create -f volume.yaml
pod/non-persistent-redis created
root@node-1:~# k get po -n staging
NAME                READY   STATUS    RESTARTS   AGE
non-persistent-redis 1/1     Running   0           6s
root@node-1:~#
```

F:\Work\Data Entry Work\Data Entry\20200827\CKA\13 D.JPG

**NEW QUESTION 21**

CORRECT TEXT

Score: 13%

Set configuration context: 

```
[student@node-1] $ | kube
ctl config use-context w
k8s
```

**Task**

A Kubernetes worker node, named wk8s-node-0 is in state NotReady. Investigate why this is the case, and perform any appropriate steps to bring the node to a Ready state, ensuring that any changes are made permanent.

You can ssh to the failed node using: 

```
[student@node-1] $ | ssh
wk8s-node-0
```

You can assume elevated privileges on the node with the following command:

```
[student@wk8s-node-0] $ |
sudo -i
```

- A. Mastered
- B. Not Mastered

**Answer:** A

**Explanation:**

Solution:

```
sudo -i
systemctl status kubelet
systemctl start kubelet
systemctl enable kubelet
```

**NEW QUESTION 25**

CORRECT TEXT

Perform the following tasks:

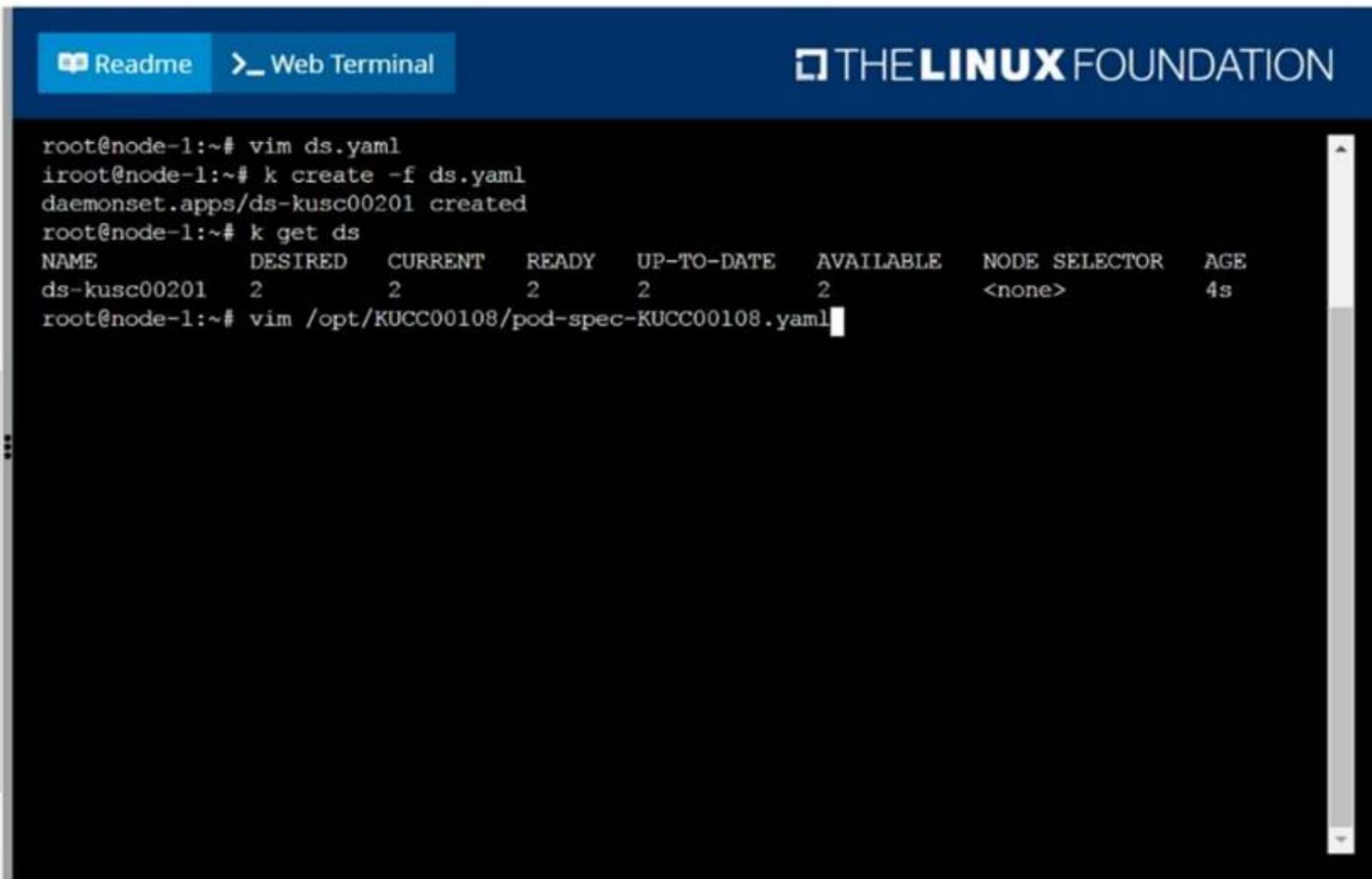
- ? Add an init container to hungry-bear (which has been defined in spec file /opt/KUCC00108/pod-spec-KUCC00108.yaml)
- ? The init container should create an empty file named /workdir/calm.txt
- ? If /workdir/calm.txt is not detected, the pod should exit
- ? Once the spec file has been updated with the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

A.

**Answer:** Seethesolutionbelow.

**Explanation:**

solution



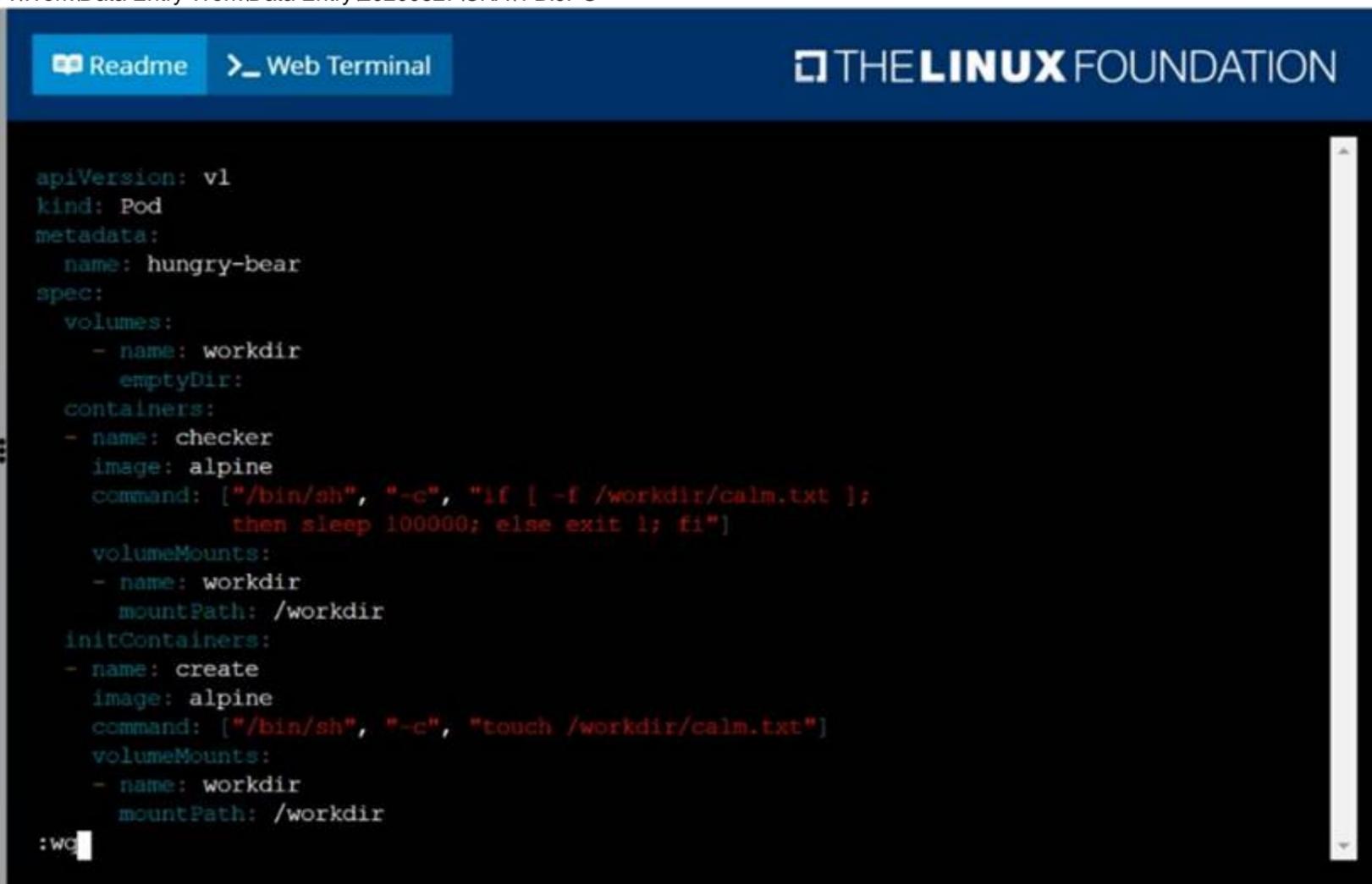
The screenshot shows a web terminal window with a dark background and a blue header. The header contains a 'Readme' button, a 'Web Terminal' button, and the 'THE LINUX FOUNDATION' logo. The terminal content shows the following sequence of commands and output:

```

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

```

F:\Work\Data Entry Work\Data Entry\20200827\CKA\4 B.JPG



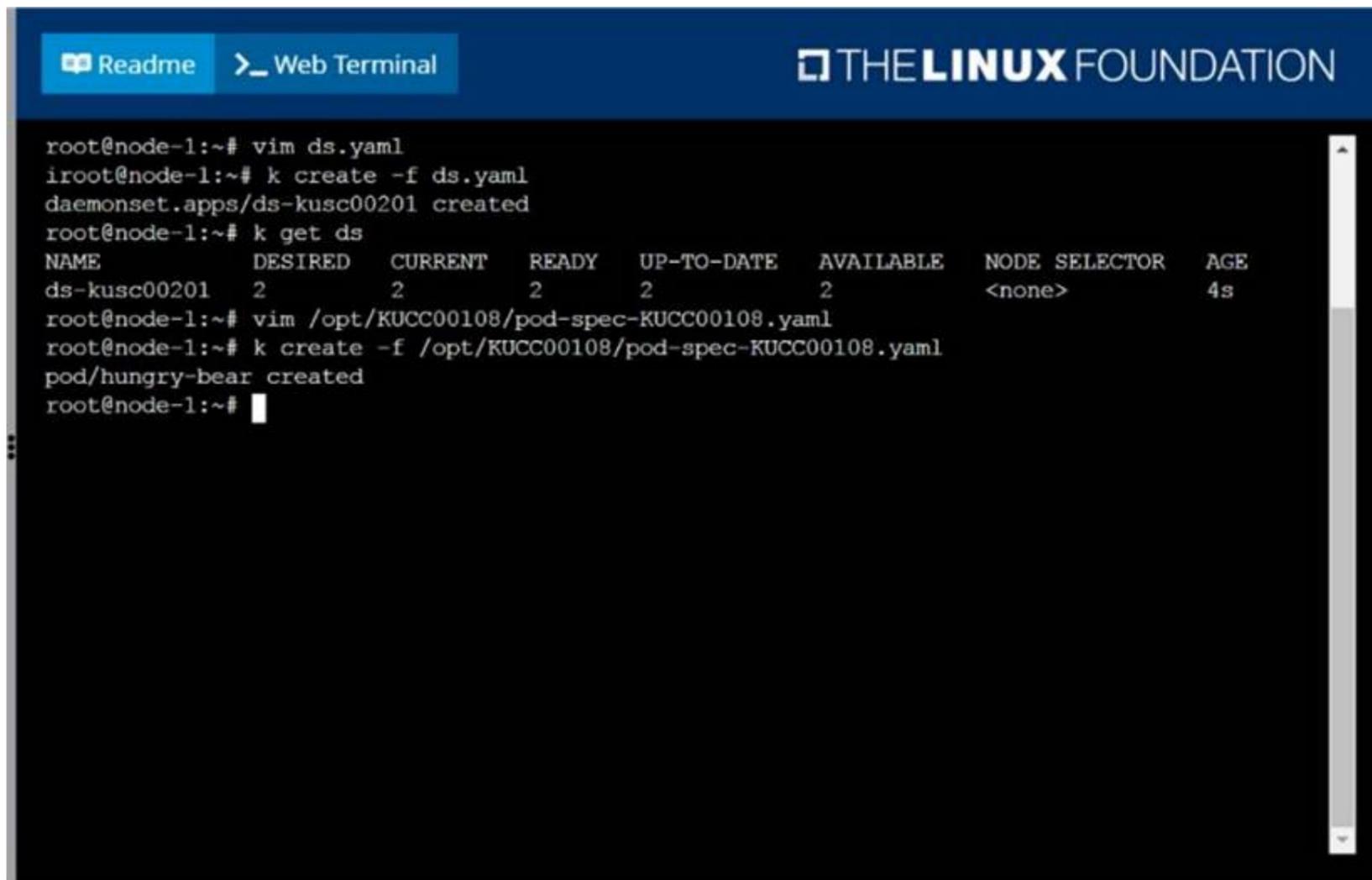
The screenshot shows a web terminal window with a dark background and a blue header. The header contains a 'Readme' button, a 'Web Terminal' button, and the 'THE LINUX FOUNDATION' logo. The terminal content shows a YAML pod specification:

```

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
:wc

```

F:\Work\Data Entry Work\Data Entry\20200827\CKA\4 C.JPG



```

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:~#

```

F:\Work\Data Entry Work\Data Entry\20200827\CKA\4 D.JPG

**NEW QUESTION 26**

CORRECT TEXT

A Kubernetes worker node, named wk8s-node-0 is in state NotReady. Investigate why this is the case, and perform any appropriate steps to bring the node to a Ready state, ensuring that any changes are made permanent.

You can ssh to the failed node using:

```
[student@node-1] $ | ssh Wk8s-node-0
```

You can assume elevated privileges on the node with the following command:

```
[student@w8ks-node-0] $ | sudo -i
```

- A. Mastered
- B. Not Mastered

**Answer:** A

**Explanation:**

solution

```

Readme Web Terminal THE LINUX FOUNDATION

root@node-1:~# kubectl config use-context wk8s
Switched to context "wk8s".
root@node-1:~# k get nodes
NAME          STATUS    ROLES    AGE   VERSION
wk8s-master-0 Ready     master   77d   v1.18.2
wk8s-node-0   NotReady <none>   77d   v1.18.2
wk8s-node-1   Ready     <none>   77d   v1.18.2
root@node-1:~# ssh wk8s-node-0

```

F:\Work\Data Entry Work\Data Entry\20200827\CKA\20 C.JPG

```

Readme Web Terminal THE LINUX FOUNDATION

wk8s-node-0   NotReady <none>   77d   v1.18.2
wk8s-node-1   Ready     <none>   77d   v1.18.2
root@node-1:~# ssh wk8s-node-0
Welcome to Ubuntu 16.04.6 LTS (GNU/Linux 4.4.0-1109-aws x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:    https://landscape.canonical.com
 * Support:       https://ubuntu.com/advantage

 * Are you ready for Kubernetes 1.19? It's nearly here! Try RC3 with
   sudo snap install microk8s --channel=1.19/candidate --classic
   https://microk8s.io/ has docs and details.

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

New release '18.04.5 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

student@wk8s-node-0:~$ sudo -i
root@wk8s-node-0:~# systemctl restart kubelet
root@wk8s-node-0:~# systemctl enable kubelet

```

F:\Work\Data Entry Work\Data Entry\20200827\CKA\20 D.JPG

Readme
Web Terminal
THE LINUX FOUNDATION

```

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

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

New release '18.04.5 LTS' available.
Run 'do-release-upgrade' to upgrade to it.

student@wk8s-node-0:~$ sudo -i
root@wk8s-node-0:~# systemctl restart kubelet
root@wk8s-node-0:~# systemctl enable kubelet
Created symlink from /etc/systemd/system/multi-user.target.wants/kubelet.service to /lib/systemd/system/kubelet.service.
root@wk8s-node-0:~# exit
logout
student@wk8s-node-0:~$ exit
logout
Connection to 10.250.5.34 closed.
root@node-1:~# k get nodes
NAME             STATUS    ROLES    AGE   VERSION
wk8s-master-0   Ready    master   77d   v1.18.2
wk8s-node-0     Ready    <none>   77d   v1.18.2
wk8s-node-1     Ready    <none>   77d   v1.18.2
root@node-1:~#

```

F:\Work\Data Entry Work\Data Entry\20200827\CKA\20 E.JPG

**NEW QUESTION 31**

CORRECT TEXT

List all the pods sorted by name

- A. Mastered
- B. Not Mastered

**Answer:** A

**Explanation:**

kubectl get pods --sort-by=.metadata.name

**NEW QUESTION 33**

CORRECT TEXT

Schedule a pod as follows:

? Name: nginx-kusc00101

? Image: nginx

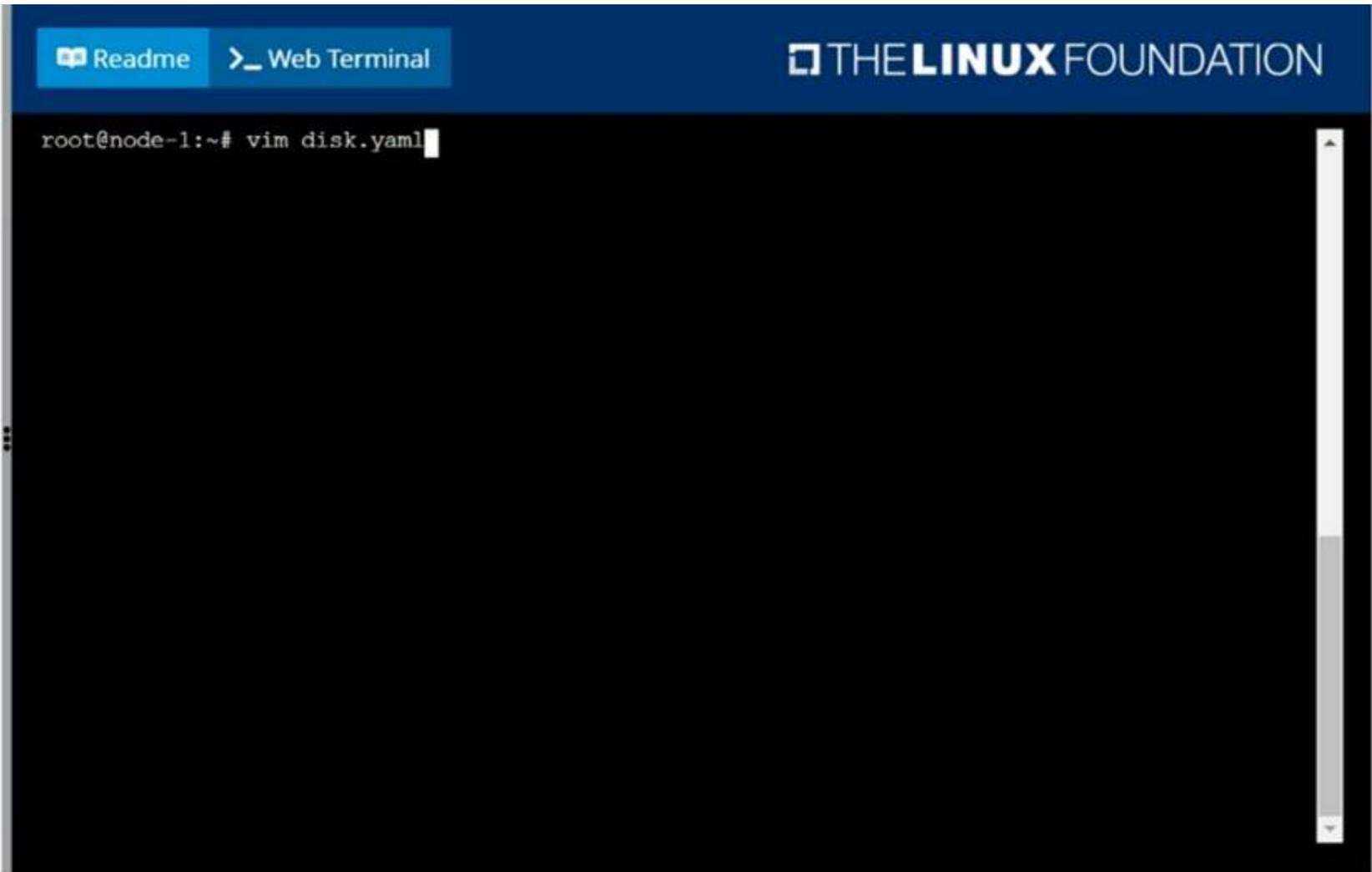
? Node selector: disk=ssd

- A. Mastered
- B. Not Mastered

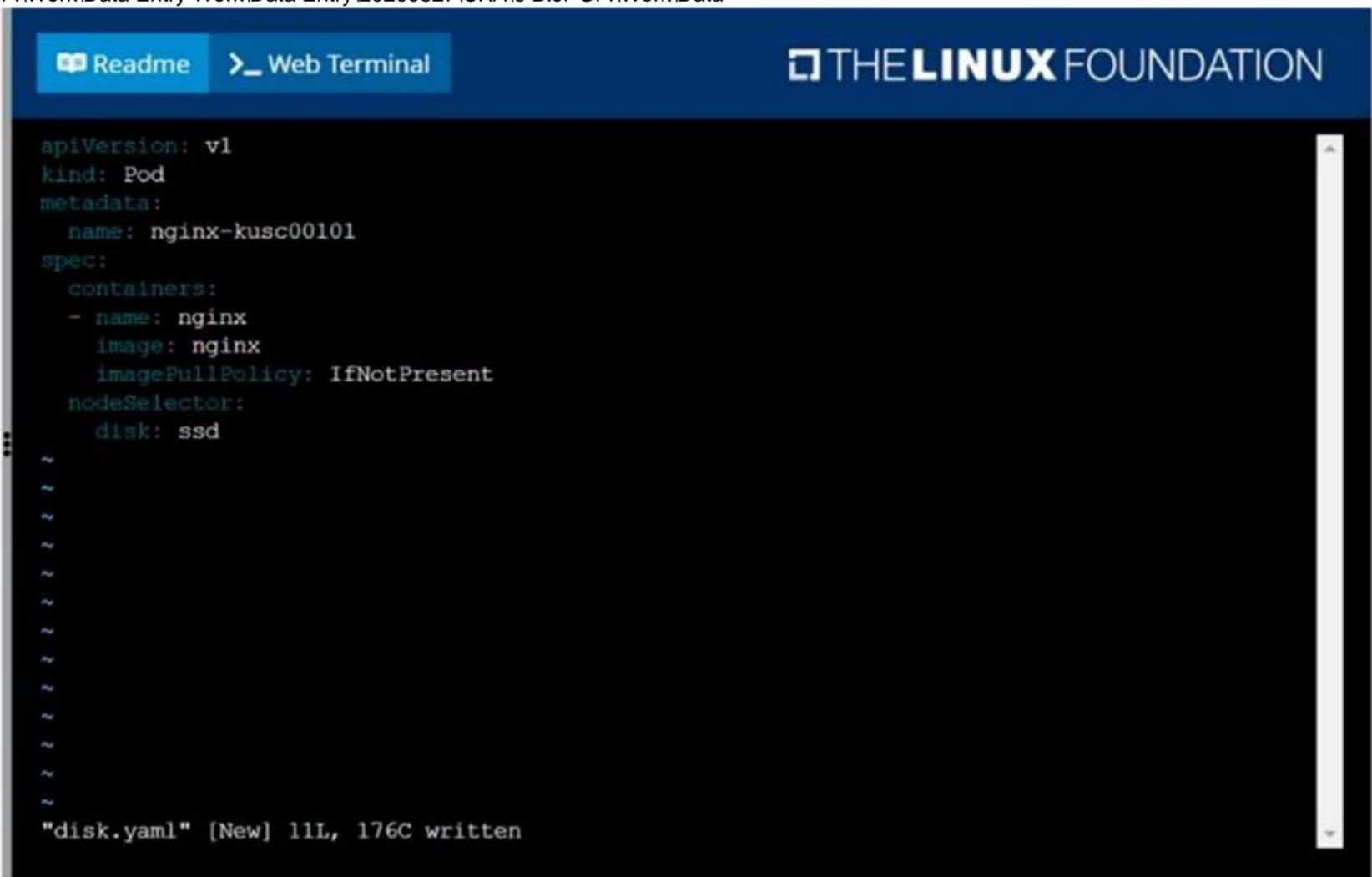
**Answer:** A

**Explanation:**

solution



F:\Work\Data Entry Work\Data Entry\20200827\CKA\6 B.JPGF:\Work\Data



F:\Work\Data Entry Work\Data Entry\20200827\CKA\6 C.JPG

Readme
Web Terminal
THE LINUX FOUNDATION

```

root@node-1:~# vim disk.yaml
root@node-1:~# k create -f disk.yaml
pod/nginx-kusc00101 created
root@node-1:~# k get po
NAME                READY   STATUS    RESTARTS   AGE
cpu-utilizer-98b9se 1/1     Running   0           5h59m
cpu-utilizer-ab2d3s 1/1     Running   0           5h59m
cpu-utilizer-kipb9a 1/1     Running   0           5h59m
ds-kusc00201-2r2k9  1/1     Running   0           13m
ds-kusc00201-hzm9q  1/1     Running   0           13m
foo                  1/1     Running   0           6h1m
front-end            1/1     Running   0           6h1m
hungry-bear          1/1     Running   0           9m37s
kucc8                3/3     Running   0           7m37s
nginx-kusc00101      1/1     Running   0           9s
webserver-84c55967f4-qzjcv 1/1     Running   0           6h16m
webserver-84c55967f4-t4791 1/1     Running   0           6h16m
root@node-1:~#

```

F:\Work\Data Entry Work\Data Entry\20200827\CKA\6 D.JPG

**NEW QUESTION 35**

CORRECT TEXT

List the nginx pod with custom columns POD\_NAME and POD\_STATUS

- A. Mastered
- B. Not Mastered

**Answer:** A

**Explanation:**

kubectl get po -o=custom-columns="POD\_NAME:.metadata.name, POD\_STATUS:.status.containerStatuses[].state"

**NEW QUESTION 36**

CORRECT TEXT

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

- A. Mastered
- B. Not Mastered

**Answer:** A

**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

```

**NEW QUESTION 39**

.....

## **Thank You for Trying Our Product**

### **We offer two products:**

1st - We have Practice Tests Software with Actual Exam Questions

2nd - Questions and Answers in PDF Format

### **CKA Practice Exam Features:**

- \* CKA Questions and Answers Updated Frequently
- \* CKA Practice Questions Verified by Expert Senior Certified Staff
- \* CKA Most Realistic Questions that Guarantee you a Pass on Your FirstTry
- \* CKA Practice Test Questions in Multiple Choice Formats and Updatesfor 1 Year

**100% Actual & Verified — Instant Download, Please Click**  
**[Order The CKA Practice Test Here](#)**