CKA Dumps

Certified Kubernetes Administrator (CKA) Program

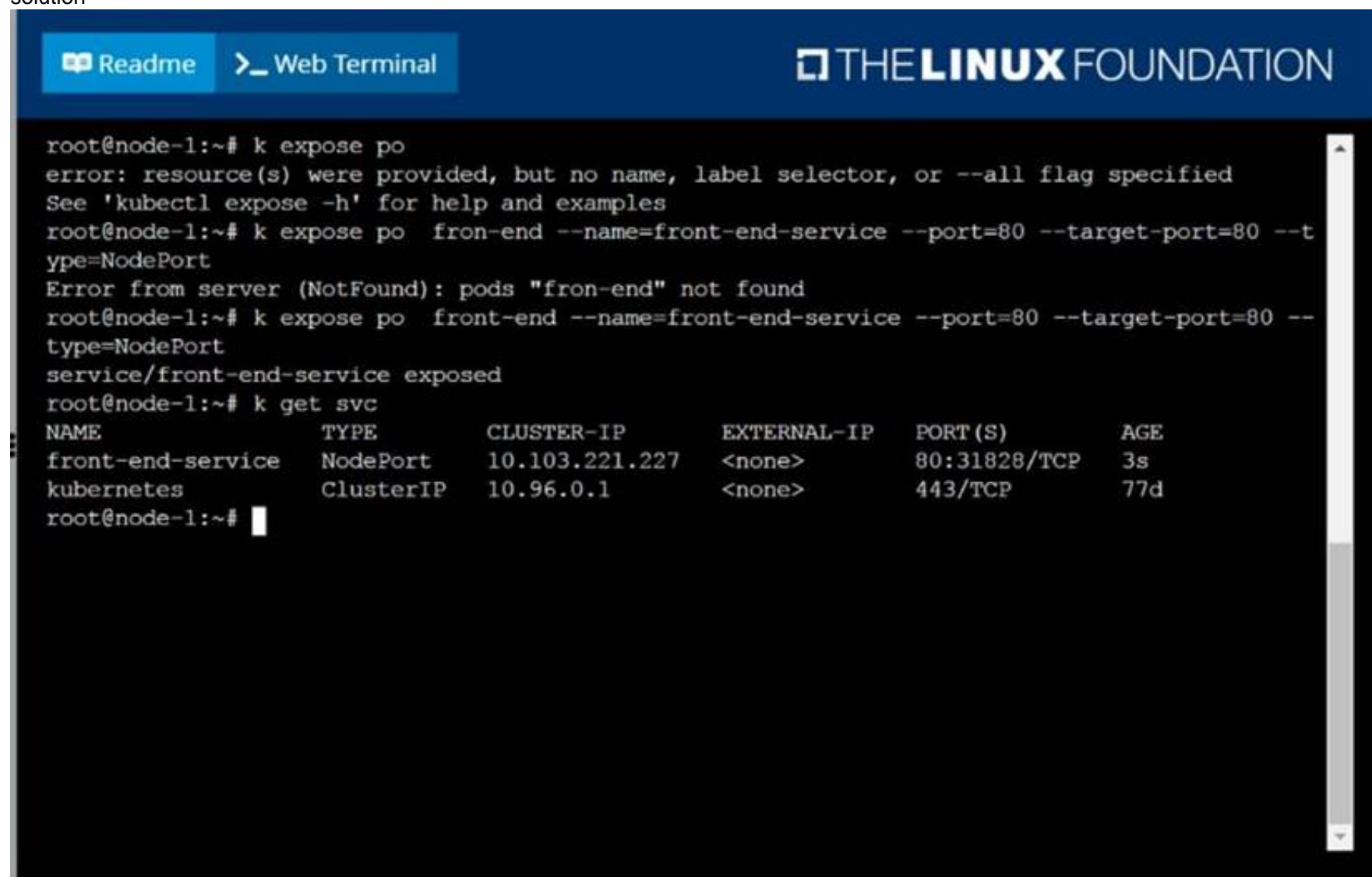# https://www.certleader.com/CKA-dumps.html

NEW QUESTION 1
CORRECT TEXT
Create and configure the service front-end-service so it's accessible through NodePort and routes to the existing pod named front-end.

A. Mastered
B. Not Mastered

Answer: A

Explanation:
solution

```
root@node-1:~# k expose po
error: resource(s) were provided, but no name, label selector, or --all flag specified
See 'kubectl expose -h' for help and examples
root@node-1:~# k expose po  fron-end --name=front-end-service --port=80 --target-port=80 --t
ype=NodePort
Error from server (NotFound): pods "fron-end" not found
root@node-1:~# k expose po  front-end --name=front-end-service --port=80 --target-port=80 --
type=NodePort
service/front-end-service exposed
root@node-1:~# k get svc
NAME                TYPE        CLUSTER-IP       EXTERNAL-IP   PORT(S)        AGE
front-end-service   NodePort    10.103.221.227   <none>        80:31828/TCP   3s
kubernetes          ClusterIP   10.96.0.1        <none>        443/TCP        77d
root@node-1:~#
```

F:\Work\Data Entry Work\Data Entry\20200827\CKA\8 B.JPG

NEW QUESTION 2
CORRECT TEXT
Get IP address of the pod – "nginx-dev"

A. Mastered
B. Not Mastered

Answer: A

Explanation:
Kubect1 get po -o wide
Using JsonPath
kubect1 get pods -o=jsonpath='{range
items[*]}{.metadata.name}{"\t"}{.status.podIP}{"\n"}{end}'

NEW QUESTION 3
CORRECT TEXT
Score: 4%

Set configuration context: ⚠

```
[student@node-1] $ | kube
ctl config use-context k
8s
```

Task
Schedule a pod as follows:
• Name: nginx-kusc00401
• Image: nginx
• Node selector: disk=ssd

A. Mastered
B. Not Mastered

Answer: A

Explanation:

Solution:
```
#yaml
apiVersion: v1
kind: Pod
metadata:
name: nginx-kusc00401
spec:
containers:
- name: nginx
image: nginx
imagePullPolicy: IfNotPresent
nodeSelector:
disk: spinning
#
kubectl create -f node-select.yaml
```
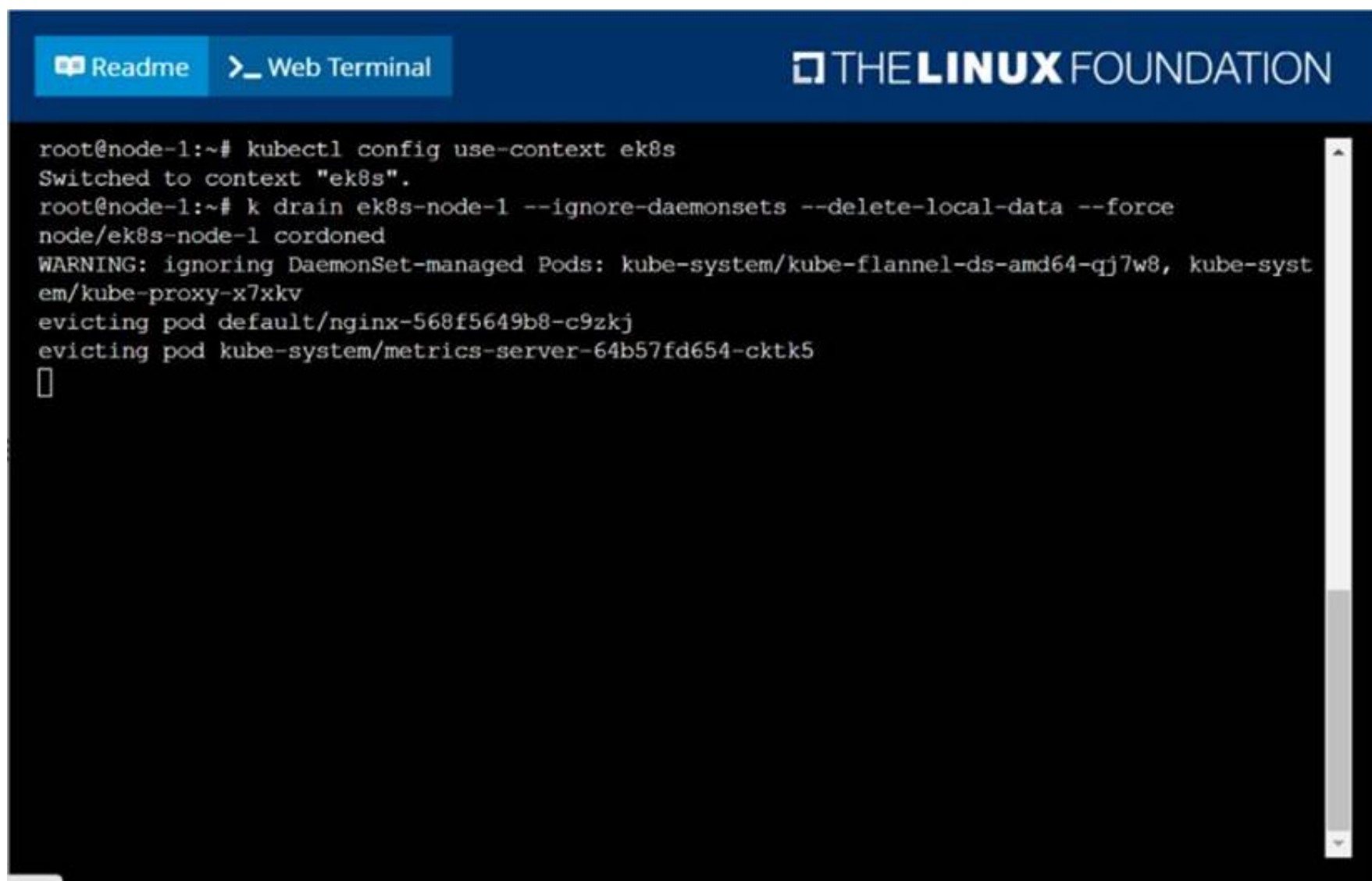
NEW QUESTION 4
CORRECT TEXT
Set the node named ek8s-node-1 as unavailable and reschedule all the pods running on it.

A. Mastered
B. Not Mastered

Answer: A

Explanation:
solution

F:\Work\Data Entry Work\Data Entry\20200827\CKA\19 B.JPG

NEW QUESTION 5
CORRECT TEXT
Scale the deployment webserver to 6 pods.

A. Mastered
B. Not Mastered

Answer: A

Explanation:
solution

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

NEW QUESTION 6

CORRECT TEXT
Monitor the logs of pod foo and:
? Extract log lines corresponding to error
unable-to-access-website
? Write them to/opt/KULM00201/foo

A. Mastered
B. Not Mastered

Answer: A

Explanation:
solution

F:\Work\Data Entry Work\Data Entry\20200827\CKA\1 B.JPG

F:\Work\Data Entry Work\Data Entry\20200827\CKA\1 C.JPG

NEW QUESTION 7

CORRECT TEXT
Score:7%

Set configuration context:

```
[student@node-1] $ | kube
ctl config use-context k
8s
```
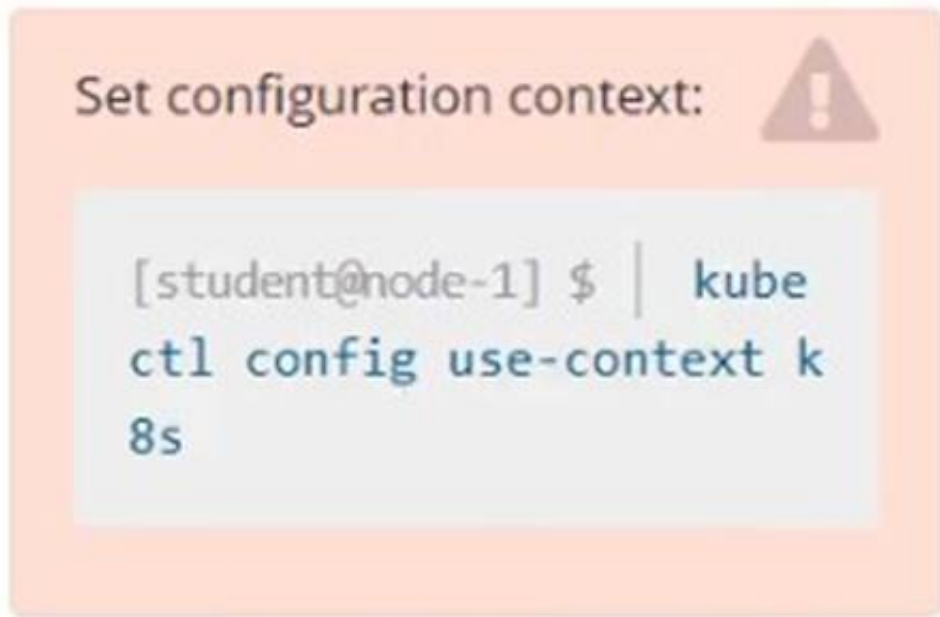
Context
An existing Pod needs to be integrated into the Kubernetes built-in logging architecture (e.

A. Mastered
B. Not Mastered

Answer: A

Explanation:

Solution:
#
kubectl get pod big-corp-app -o yaml
#
apiVersion: v1
kind: Pod
metadata:
name: big-corp-app
spec:
containers:
- name: big-corp-app
image: busybox
args:
- /bin/sh
- -c
- > i=0;
while true;
do
echo "$(date) INFO $i" >> /var/log/big-corp-app.log;
i=$((i+1));
sleep 1;
done
volumeMounts:
- name: logs
mountPath: /var/log
- name: count-log-1
image: busybox
args: [/bin/sh, -c, 'tail -n+1 -f /var/log/big-corp-app.log']
volumeMounts:
- name: logs
mountPath: /var/log
volumes:
- name: logs
emptyDir: {
}
#
kubectl logs big-corp-app -c count-log-1

NEW QUESTION 8
CORRECT TEXT
Score: 4%

Task
Scale the deployment presentation to 6 pods.

A. Mastered
B. Not Mastered

Answer: A

Explanation:
Solution:
kubectl get deployment
kubectl scale deployment.apps/presentation --replicas=6

NEW QUESTION 9
CORRECT TEXT
Score: 4%

Task
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 .

A. Mastered
B. Not Mastered

Answer: A

Explanation:

Solution:
kubectl run kucc8 --image=nginx --dry-run -o yaml > kucc8.yaml
vi kucc8.yaml
apiVersion: v1
kind: Pod
metadata:
creationTimestamp: null
name: kucc8
spec:
containers:
- image: nginx
name: nginx
- image: redis
name: redis
- image: memcached
name: memcached
- image: consul

name: consul
#
kubectl create -f kucc8.yaml
#12.07

NEW QUESTION 10
CORRECT TEXT
Task Weight: 4%


```
[student@node-1] $   kubectl config
use-context k8s
```

Task
Scale the deployment webserver to 3 pods.

A. Mastered
B. Not Mastered

Answer: A

Explanation:
 Solution:


```
student@node-1:~$ kubectl  scale  deploy webserver  --replicas=3
deployment.apps/webserver scaled
student@node-1:~$ kubectl  scale  deploy webserver  --replicas=3
```

NEW QUESTION 10
CORRECT TEXT
Get list of all pods in all namespaces and write it to file "/opt/pods-list.yaml"

A. Mastered
B. Not Mastered

Answer: A

Explanation:
kubectl get po –all-namespaces > /opt/pods-list.yaml

NEW QUESTION 15
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
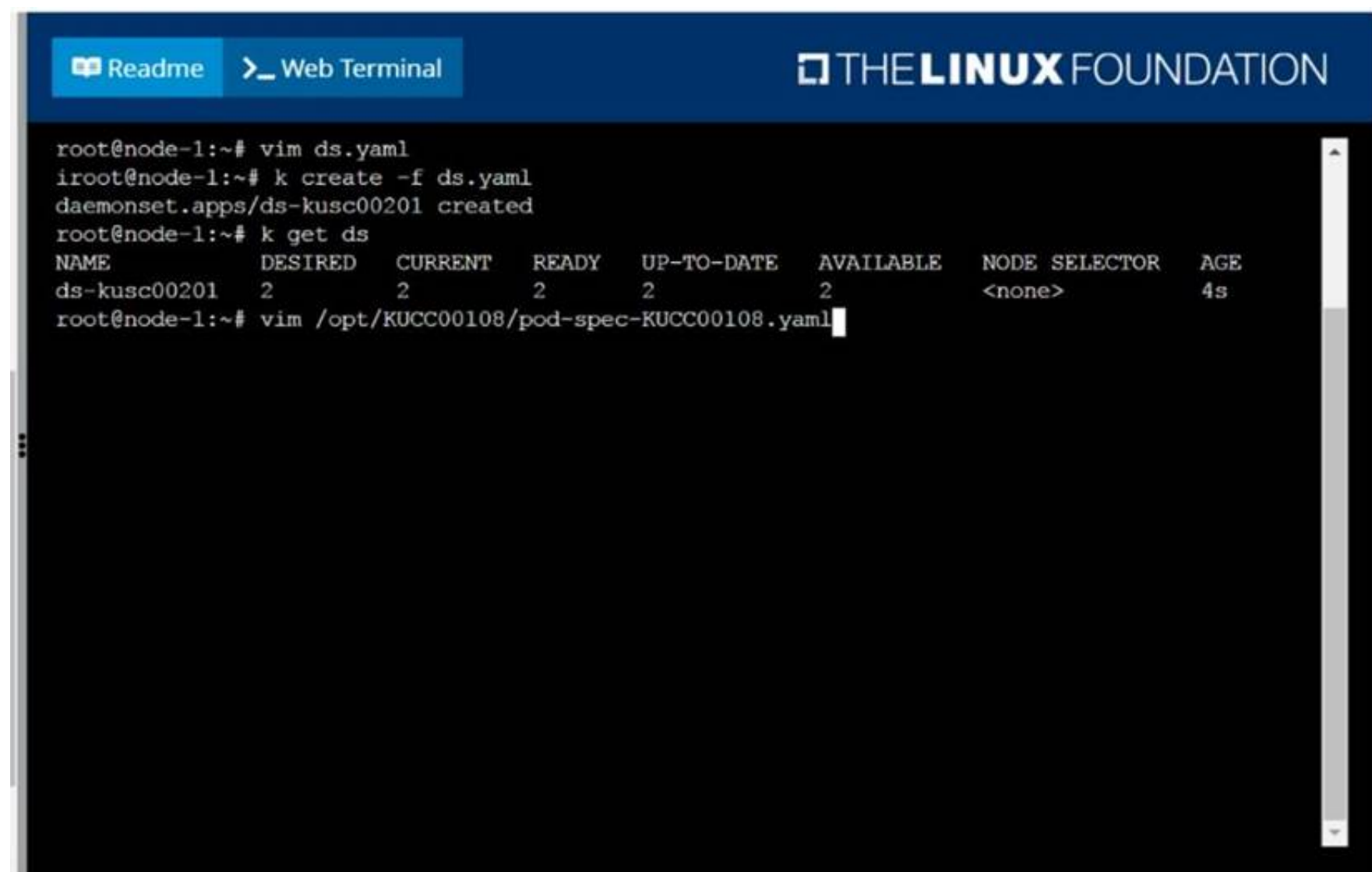
A.

Answer: Seethesolutionbelow.

Explanation:
solution
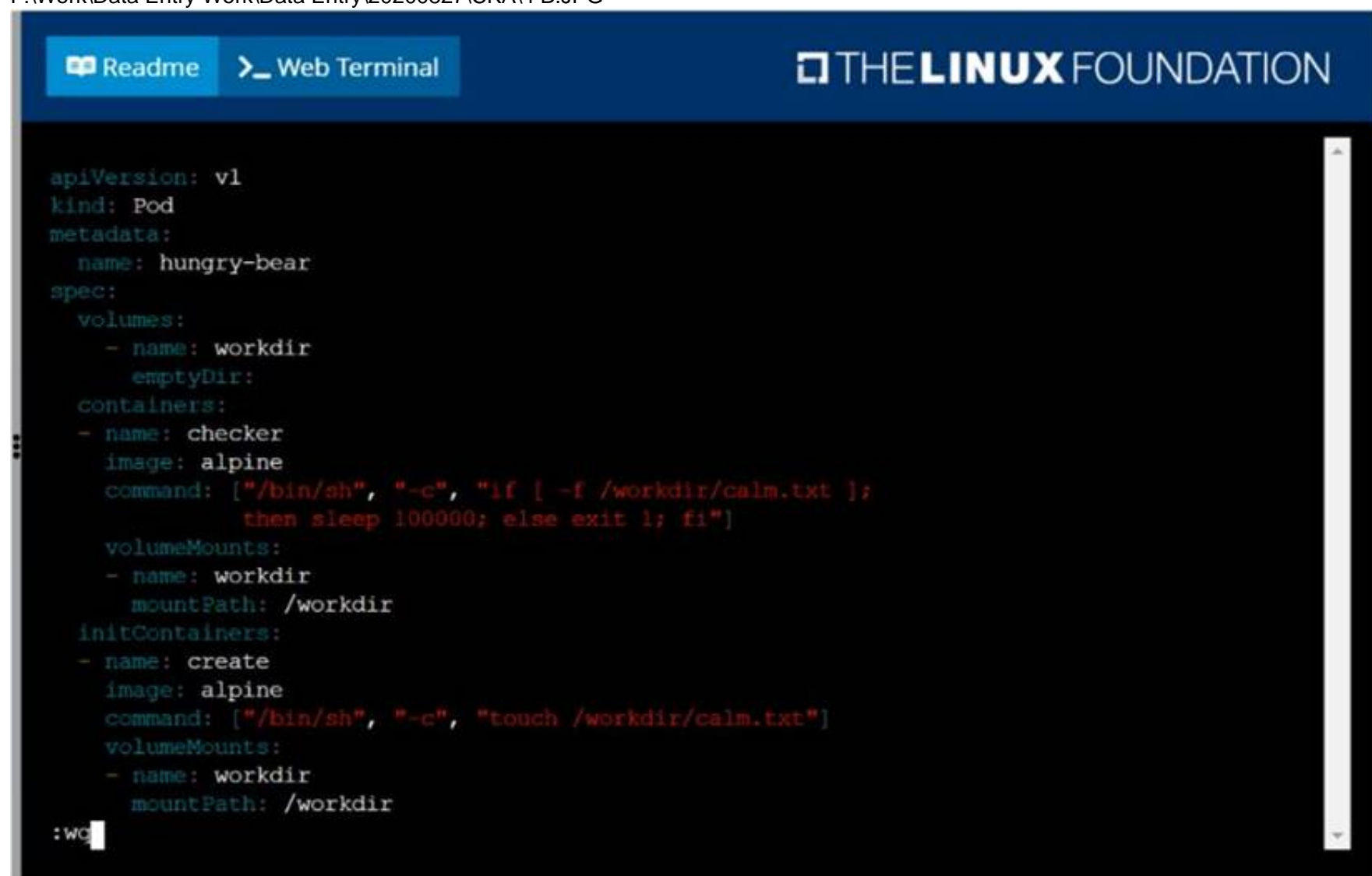
📖 Readme >_ Web Terminal

🔲 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
```

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

📖 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
```

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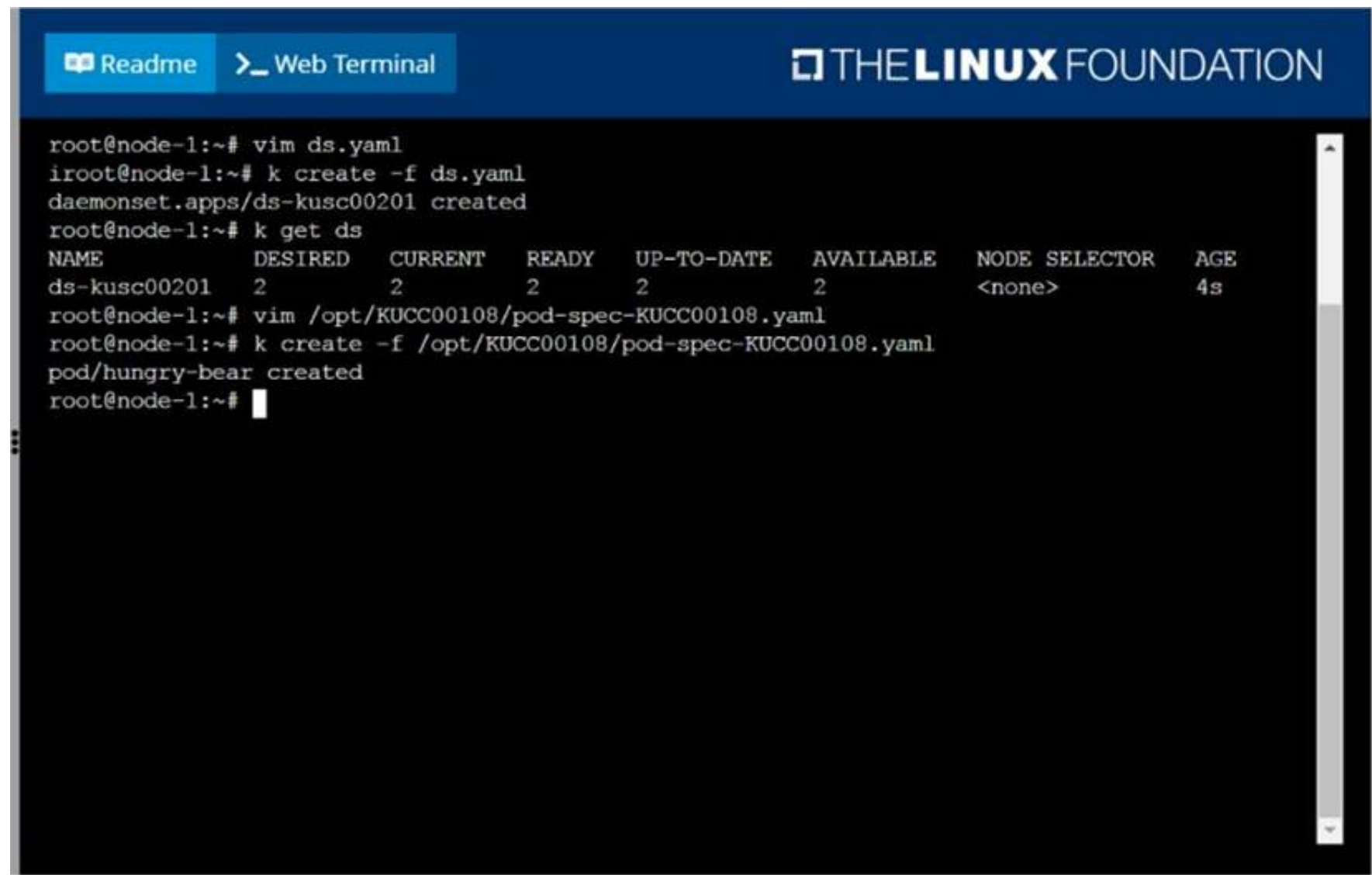
F:\Work\Data Entry Work\Data Entry\20200827\CKA\4 D.JPG

NEW QUESTION 20
CORRECT TEXT
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.

A. Mastered
B. Not Mastered

Answer: A

Explanation:
solution

F:\Work\Data Entry Work\Data Entry\20200827\CKA\2 C.JPG

NEW QUESTION 22
CORRECT TEXT
Create a persistent volume with name app-data, of capacity 2Gi and access mode ReadWriteMany. The type of volume is hostPath and its location is /srv/app-data.

A. Mastered
B. Not Mastered

Answer: A

Explanation:
solution
Persistent Volume
A persistent volume is a piece of storage in a Kubernetes cluster. PersistentVolumes are a cluster-level resource like nodes, which don't belong to any namespace. It is provisioned by the administrator and has a particular file size. This way, a developer deploying their app on Kubernetes need not know the underlying infrastructure. When the developer needs a certain amount of persistent storage for their application, the system administrator configures the cluster so that they consume the PersistentVolume provisioned in an easy way.
Creating Persistent Volume
kind: PersistentVolumeapiVersion: v1metadata: name:app-dataspec: capacity: # defines the capacity of PV we are creating storage: 2Gi #the amount of storage we are tying to claim accessModes: # defines the rights of the volume we are creating - ReadWriteMany hostPath: path: "/srv/app-data" # path to which we are creating the volume
Challenge
? Create a Persistent Volume named app-data, with access mode ReadWriteMany, storage classname shared, 2Gi of storage capacity and the host path /srv/app-data.

```
apiVersion: v1
kind: PersistentVolume
metadata:
  name: app-data
spec:
  capacity:
    storage: 2Gi
  accessModes:
    - ReadWriteMany
  hostPath:
    path: /srv/app-data
  storageClassName: shared
~
~
~
~
~
~
~
~
~
~
~
"app-data.yaml" 12L, 194C
```

* 2. Save the file and create the persistent volume.

```
njerry191@cloudshell:~ (extreme-clone-265411)$ kubectl create -f pv.yaml
persistentvolume/pv created
```
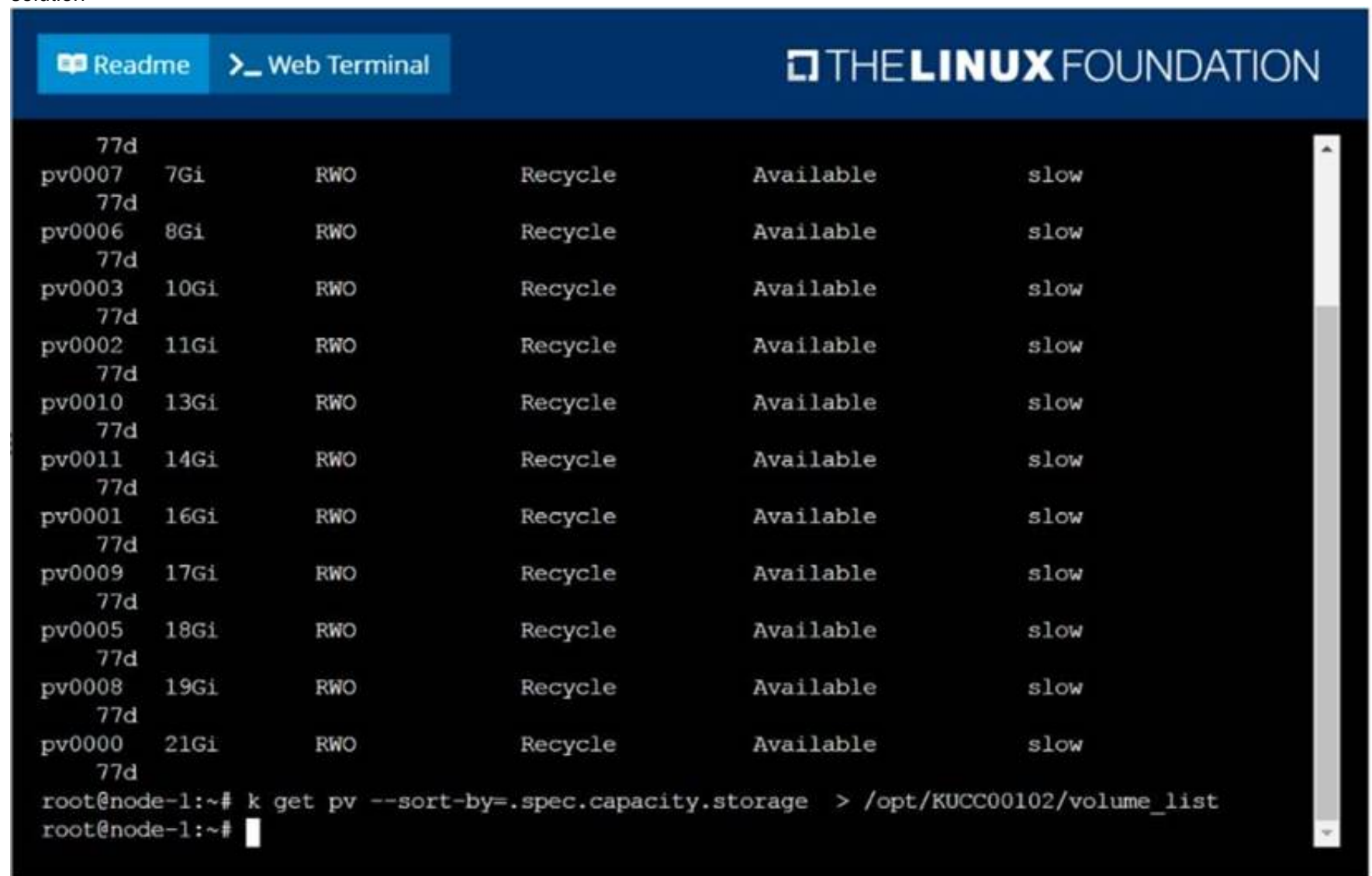
Image for post
* 3. View the persistent volume.

```
njerry191@cloudshell:~ (extreme-clone-265411)$ kubectl get pv
NAME       CAPACITY   ACCESS MODES   RECLAIM POLICY   STATUS      CLAIM   STORAGECLASS   REASON   AGE
app-data   2Gi        RWX            Retain           Available           shared                  31s
```

? Our persistent volume status is available meaning it is available and it has not been mounted yet. This status will change when we mount the persistentVolume to a persistentVolumeClaim.
PersistentVolumeClaim
In a real ecosystem, a system admin will create the PersistentVolume then a developer will create a PersistentVolumeClaim which will be referenced in a pod. A PersistentVolumeClaim is created by specifying the minimum size and the access mode they require from the persistentVolume. Challenge
? Create a Persistent Volume Claim that requests the Persistent Volume we had created above. The claim should request 2Gi. Ensure that the Persistent Volume Claim has the same storageClassName as the persistentVolume you had previously created.
kind: PersistentVolumeapiVersion: v1metadata: name:app-data
spec:
accessModes: - ReadWriteMany resources:
requests: storage: 2Gi
storageClassName: shared
* 2. Save and create the pvc
njerry191@cloudshell:~ (extreme-clone-2654111)$ kubect1 create -f app-data.yaml persistentvolumeclaim/app-data created
* 3. View the pvc

```
njerry191@cloudshell:~ (extreme-clone-265411)$ kubectl get pvc
NAME   STATUS   VOLUME   CAPACITY   ACCESS MODES   STORAGECLASS
pv     Bound    pv       512m       RWX            shared
```

Image for post
* 4. Let's see what has changed in the pv we had initially created.

```
njerry191@cloudshell:~ (extreme-clone-265411)$ kubectl get pv
NAME   CAPACITY   ACCESS MODES   RECLAIM POLICY   STATUS   CLAIM        STORAGECLASS   REASON   AGE
pv     512m       RWX            Retain           Bound    default/pv   shared                  16m
```

Image for post
Our status has now changed from available to bound.
* 5. Create a new pod named myapp with image nginx that will be used to Mount the Persistent Volume Claim with the path /var/app/config.
Mounting a Claim
apiVersion: v1kind: Podmetadata: creationTimestamp: null name: app-dataspec: volumes: - name:congigpvc persistenVolumeClaim: claimName: app-data
containers: - image: nginx name: app volumeMounts: - mountPath: "/srv/app-data " name: configpvc

NEW QUESTION 26
CORRECT TEXT
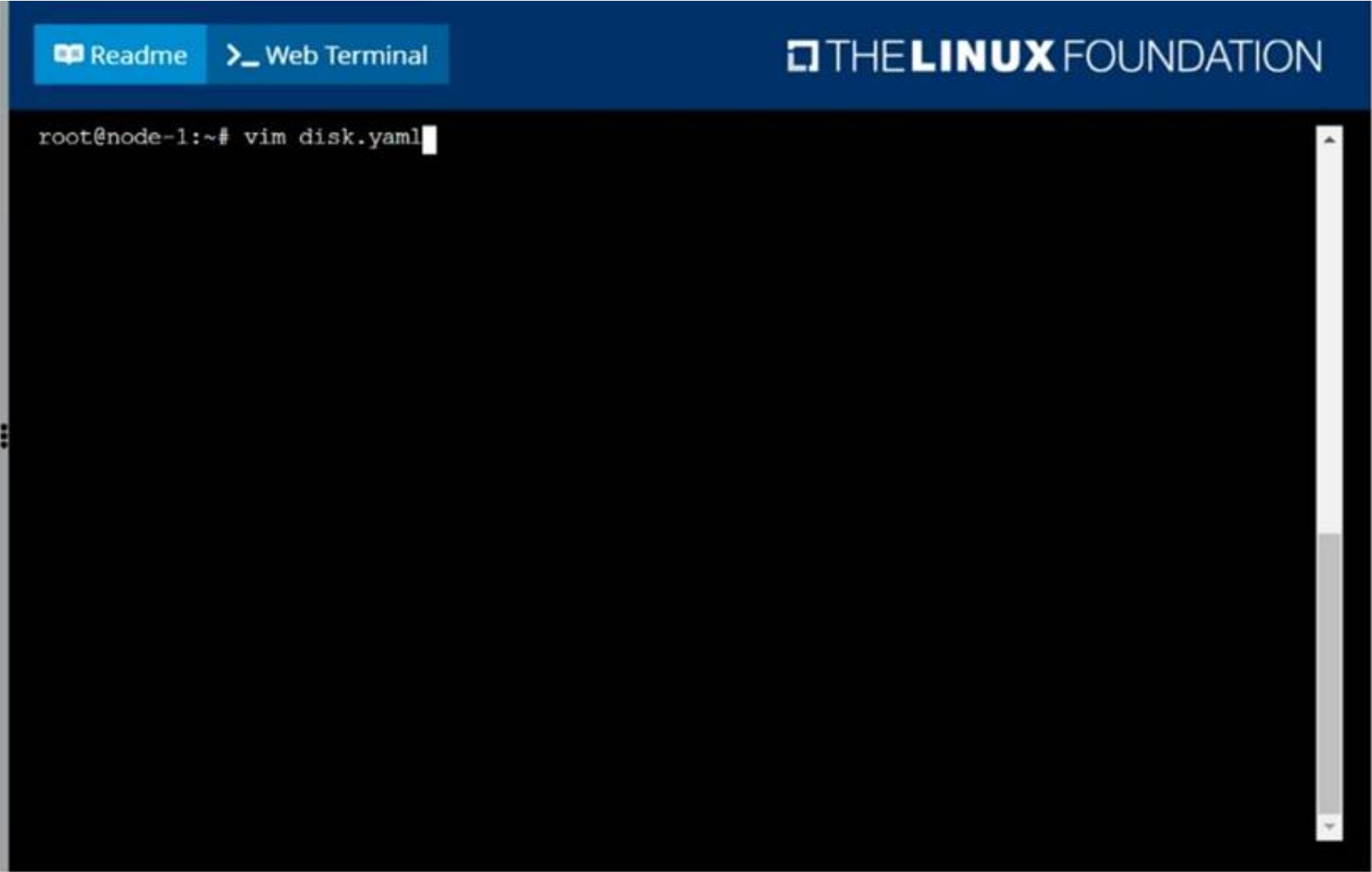Schedule a pod as follows:
? Name: nginx-kusc00101
? Image: nginx
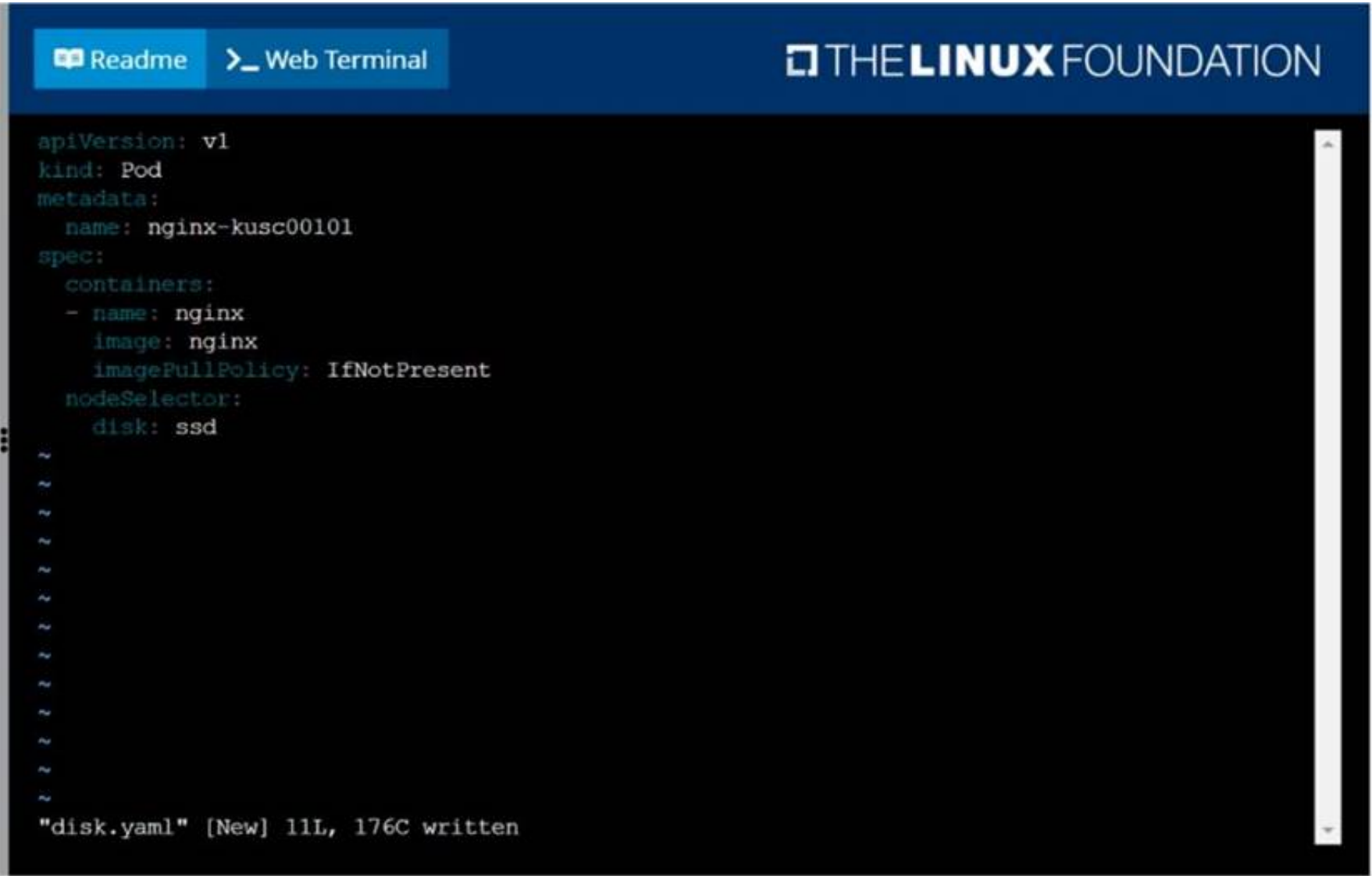? Node selector: disk=ssd

A. Mastered
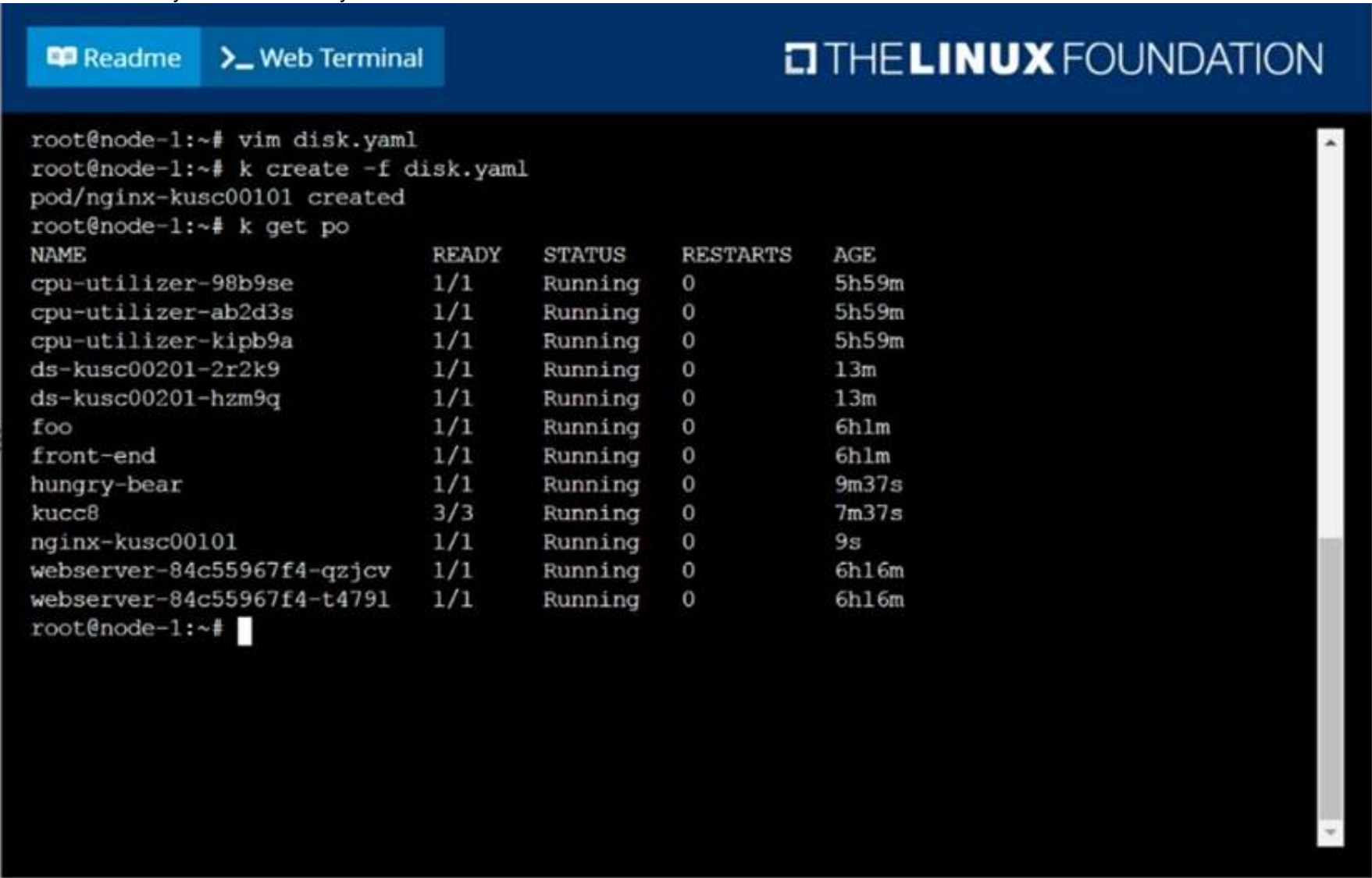B. Not Mastered

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

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

NEW QUESTION 29
CORRECT TEXT
Create a busybox pod that runs the command "env" and save the output to "envpod" file

A. Mastered
B. Not Mastered

Answer: A

Explanation:
kubectl run busybox --image=busybox --restart=Never ---rm -it -- env > envpod.yaml

NEW QUESTION 32

CORRECT TEXT
Score: 5%

Set configuration context:

```
[student@node-1] $ | kube
ctl config use-context k
8s
```

Task
Monitor the logs of pod bar and:
• Extract log lines corresponding to error file-not-found
• Write them to /opt/KUTR00101/bar

A. Mastered
B. Not Mastered

Answer: A

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

NEW QUESTION 34
CORRECT TEXT
List all the pods sorted by created timestamp

A. Mastered
B. Not Mastered

Answer: A

Explanation:
kubect1 get pods--sort-by=.metadata.creationTimestamp

NEW QUESTION 38
CORRECT TEXT
Score: 4%

Set configuration context:

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

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 .

A. Mastered
B. Not Mastered

Answer: A

Explanation:

Solution:
#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

NEW QUESTION 42
CORRECT TEXT
Create 2 nginx image pods in which one of them is labelled with env=prod and another one
labelled with env=dev and verify the same.

A. Mastered
B. Not Mastered

Answer: A

Explanation:
kubectl run --generator=run-pod/v1 --image=nginx -- labels=env=prod nginx-prod --dry-run
-o yaml > nginx-prodpod.yaml Now, edit nginx-prod-pod.yaml file and remove entries like "creationTimestamp: null" "dnsPolicy: ClusterFirst"
vim nginx-prod-pod.yaml
apiVersion: v1
kind: Pod
metadata:
labels:
env: prod
name: nginx-prod
spec:
containers:
- image: nginx
name: nginx-prod
restartPolicy: Always
kubectl create -f nginx-prod-pod.yaml
kubectl run --generator=run-pod/v1 --image=nginx --
labels=env=dev nginx-dev --dry-run -o yaml > nginx-dev-pod.yaml
apiVersion: v1
kind: Pod
metadata:
labels:
env: dev
name: nginx-dev
spec:
containers:
- image: nginx
name: nginx-dev
restartPolicy: Always
kubectl create -f nginx-prod-dev.yaml
Verify :
kubectl get po --show-labels
kubectl get po -l env=prod
kubectl get po -l env=dev

NEW QUESTION 43
......

Thank You for Trying Our Product

* 100% Pass or Money Back

    All our products come with a 90-day Money Back Guarantee.

* One year free update

    You can enjoy free update one year. 24x7 online support.

* Trusted by Millions

    We currently serve more than 30,000,000 customers.

* Shop Securely

    All transactions are protected by VeriSign!

100% Pass Your CKA Exam with Our Prep Materials Via below:

https://www.certleader.com/CKA-dumps.html