# Free Questions for CKA by dumpssheet

## Shared by Bishop on 05-09-2022

**For More Free Questions and Preparation Resources**

**Check the Links on Last Page**

# Question 1

**Question Type:** MultipleChoice

SIMULATION

Schedule a pod as follows:

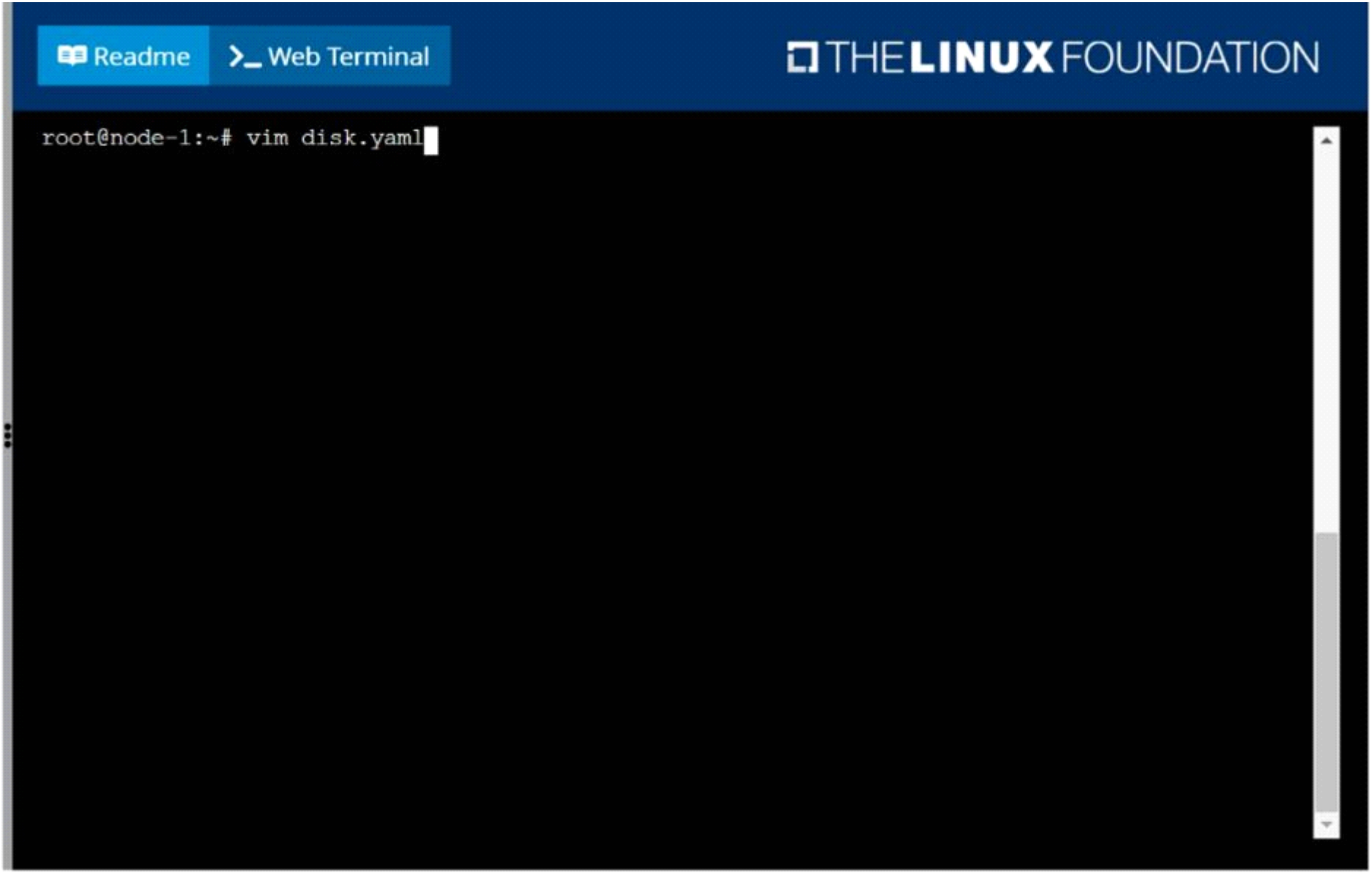
* Name: nginx-kusc00101

* Image: nginx

* Node selector: disk=ssd

## Options:

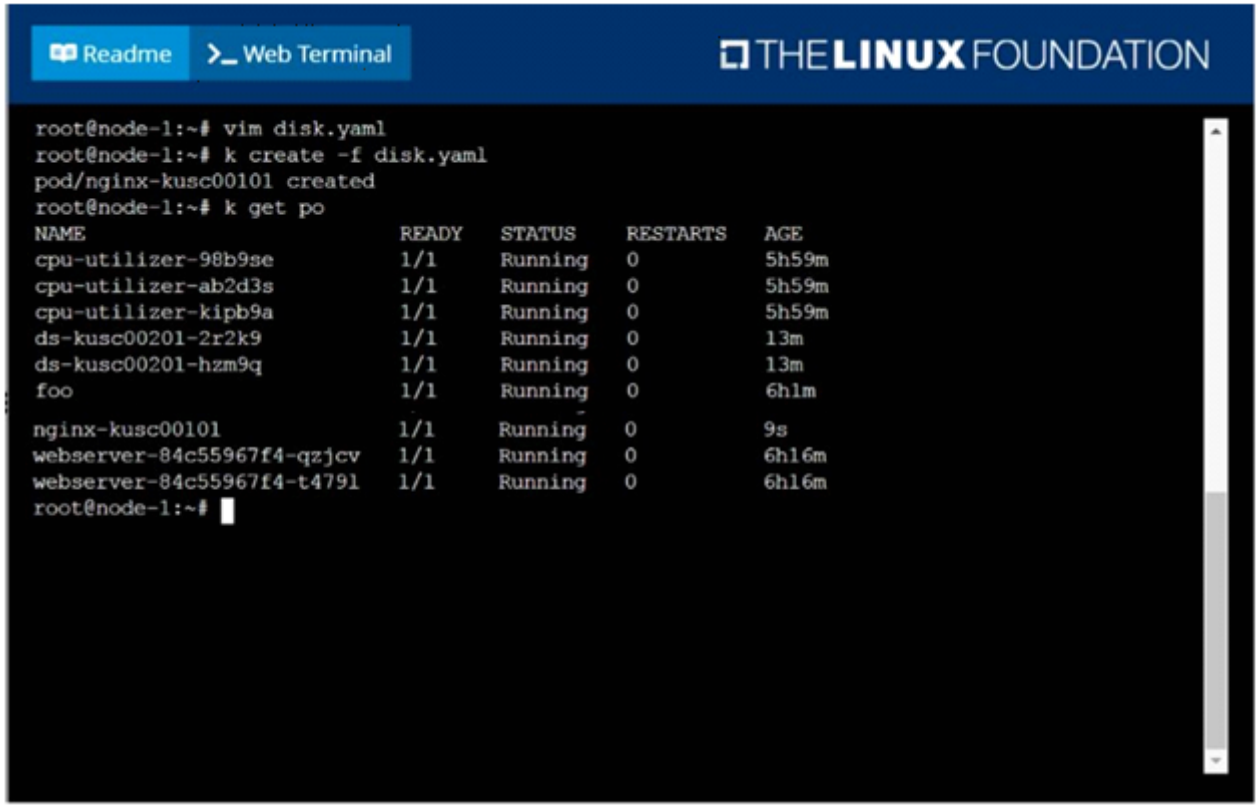
**A)** solution


Readme
Web Terminal
THE LINUX FOUNDATION
root@node-1:~# vim disk.yaml

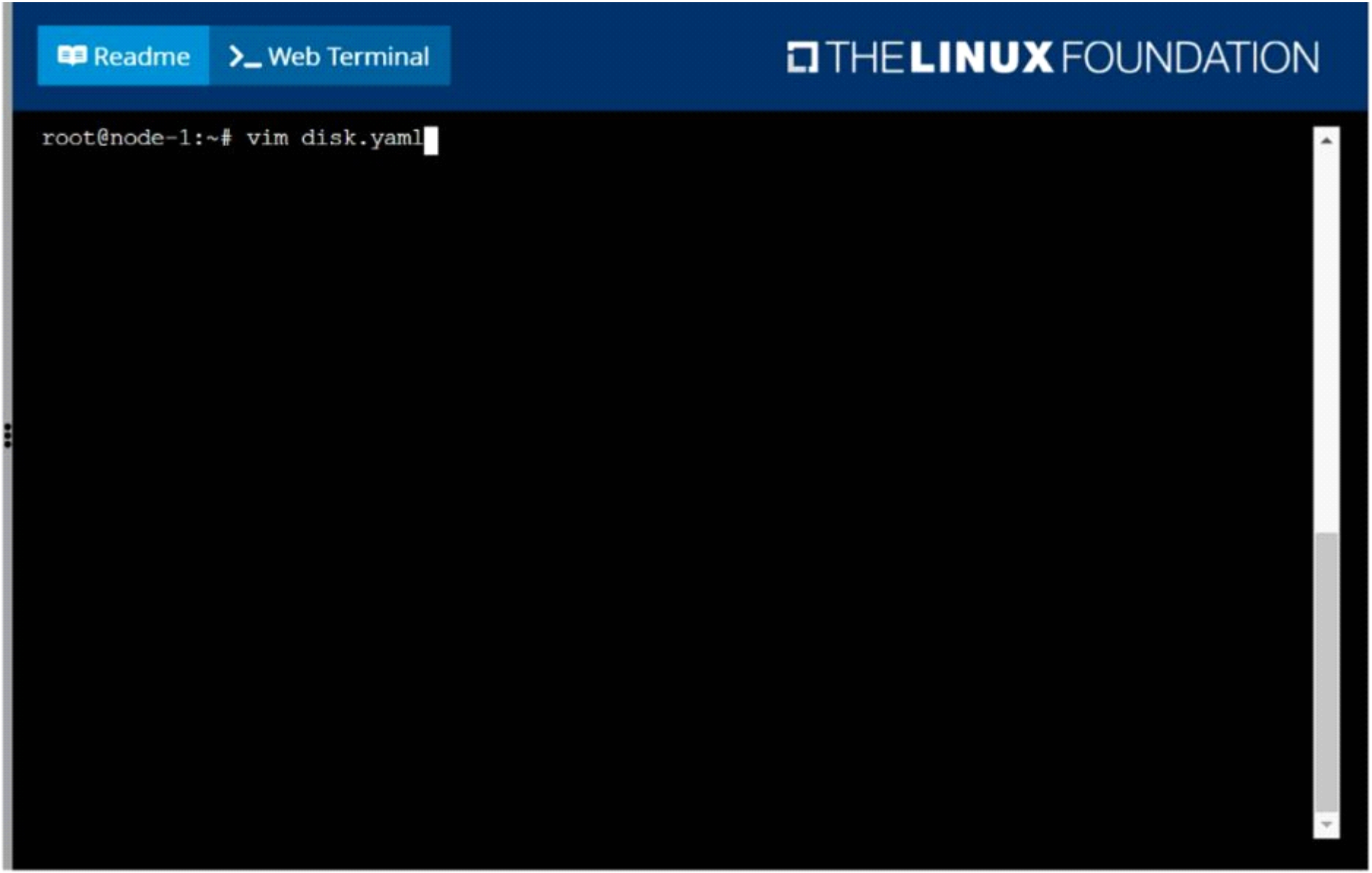
Readme | Web Terminal

THE LINUX FOUNDATION

```
apiVersion: v1
kind: Pod
metadata:
  name: nginx-kusc00101
spec:
  containers:
  - name: nginx
    image: nginx
    imagePullPolicy: IfNotPresent
  nodeSelector:
    disk: ssd
~
~
~
~
~
~
~
~
~
~
~
~
~
"disk.yaml" [New] 11L, 176C written
```


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
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:~#

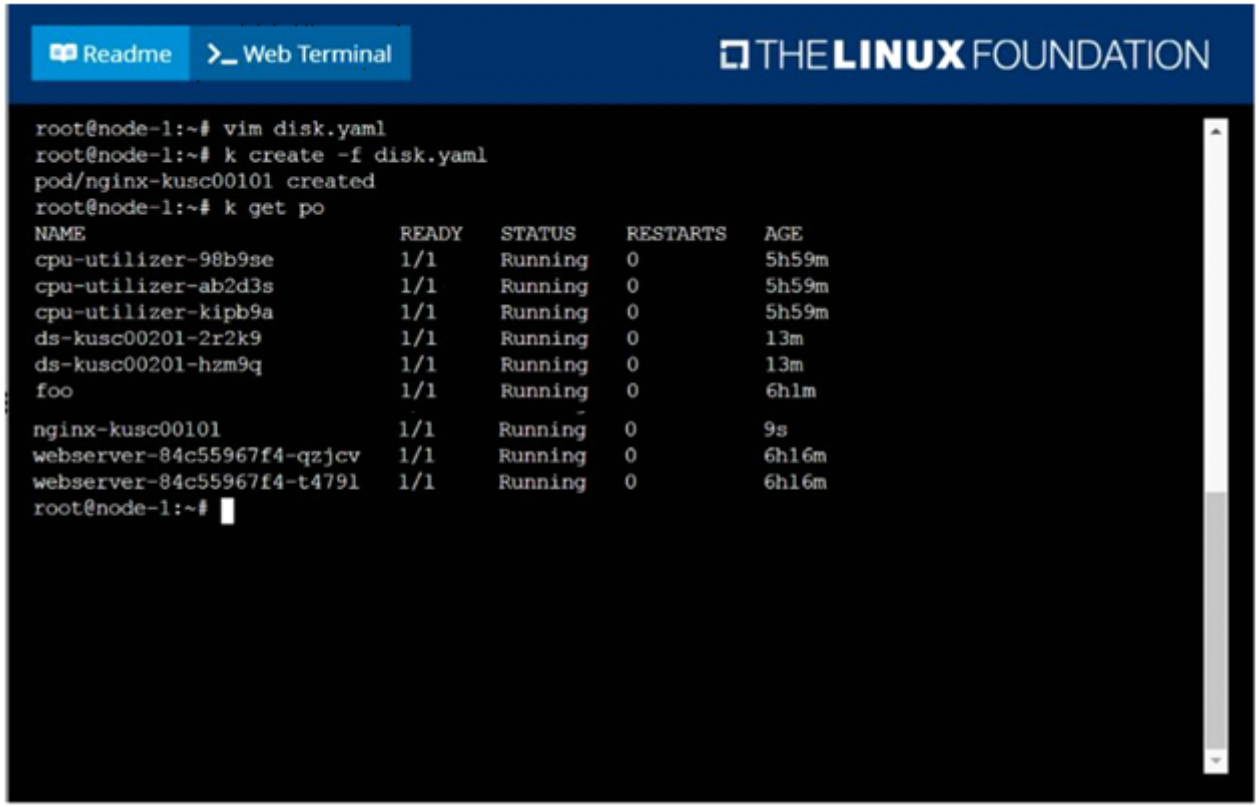
**B)** solution


Readme
Web Terminal
THE LINUX FOUNDATION
root@node-1:~# vim disk.yaml

Readme | Web Terminal

THE LINUX FOUNDATION

```
apiVersion: v1
kind: Pod
metadata:
  name: nginx-kusc00101
spec:
  containers:
  - name: nginx
    image: nginx
    imagePullPolicy: IfNotPresent
  nodeSelector:
    disk: ssd
~
~
~
~
~
~
~
~
~
~
~
~
~
"disk.yaml" [New] 11L, 176C written
```




### Answer:

A

## Question 2

**Question Type:** MultipleChoice

SIMULATION

Create a Kubernetes secret as follows:

* Name: super-secret

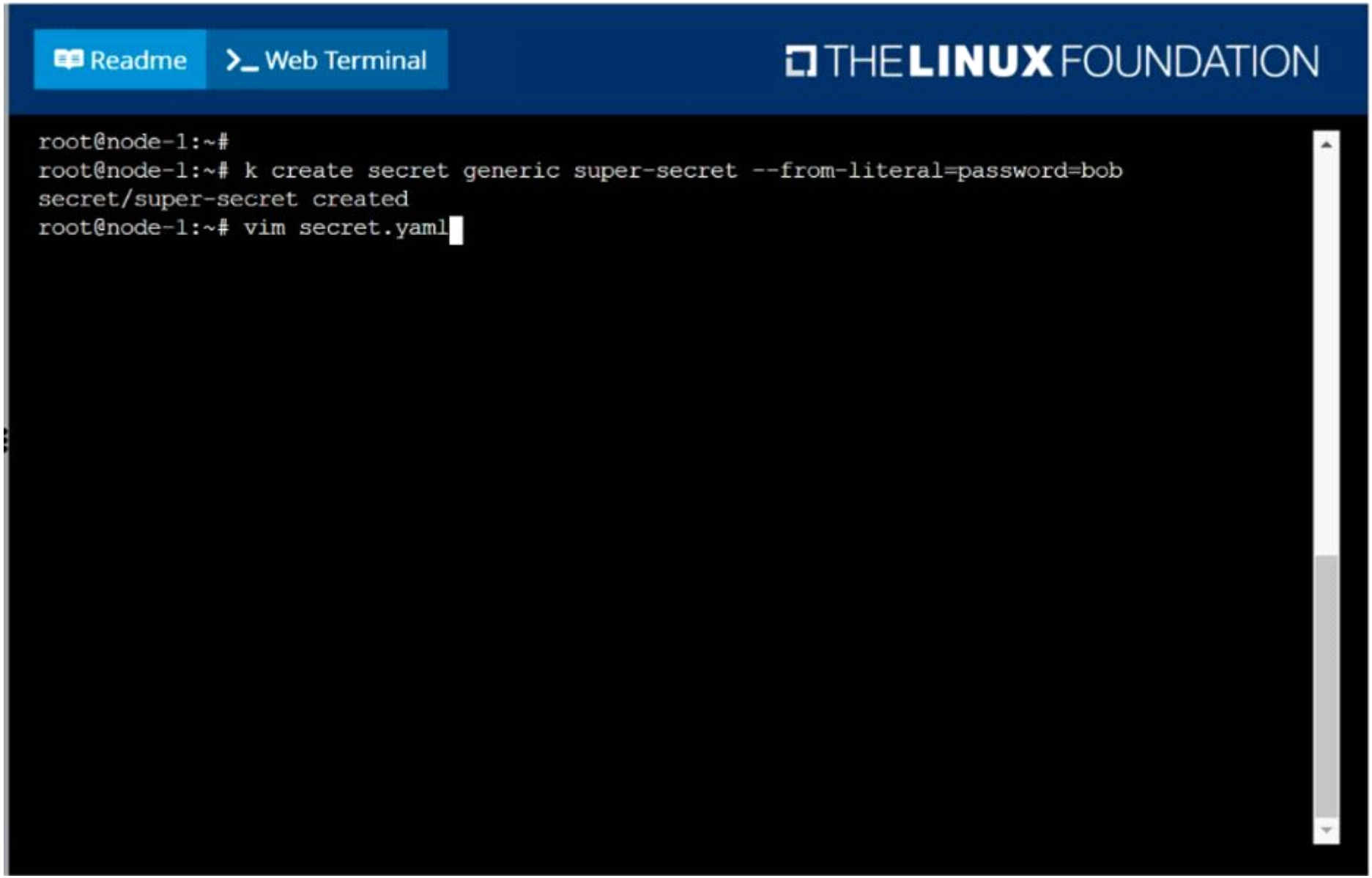
* password: bob

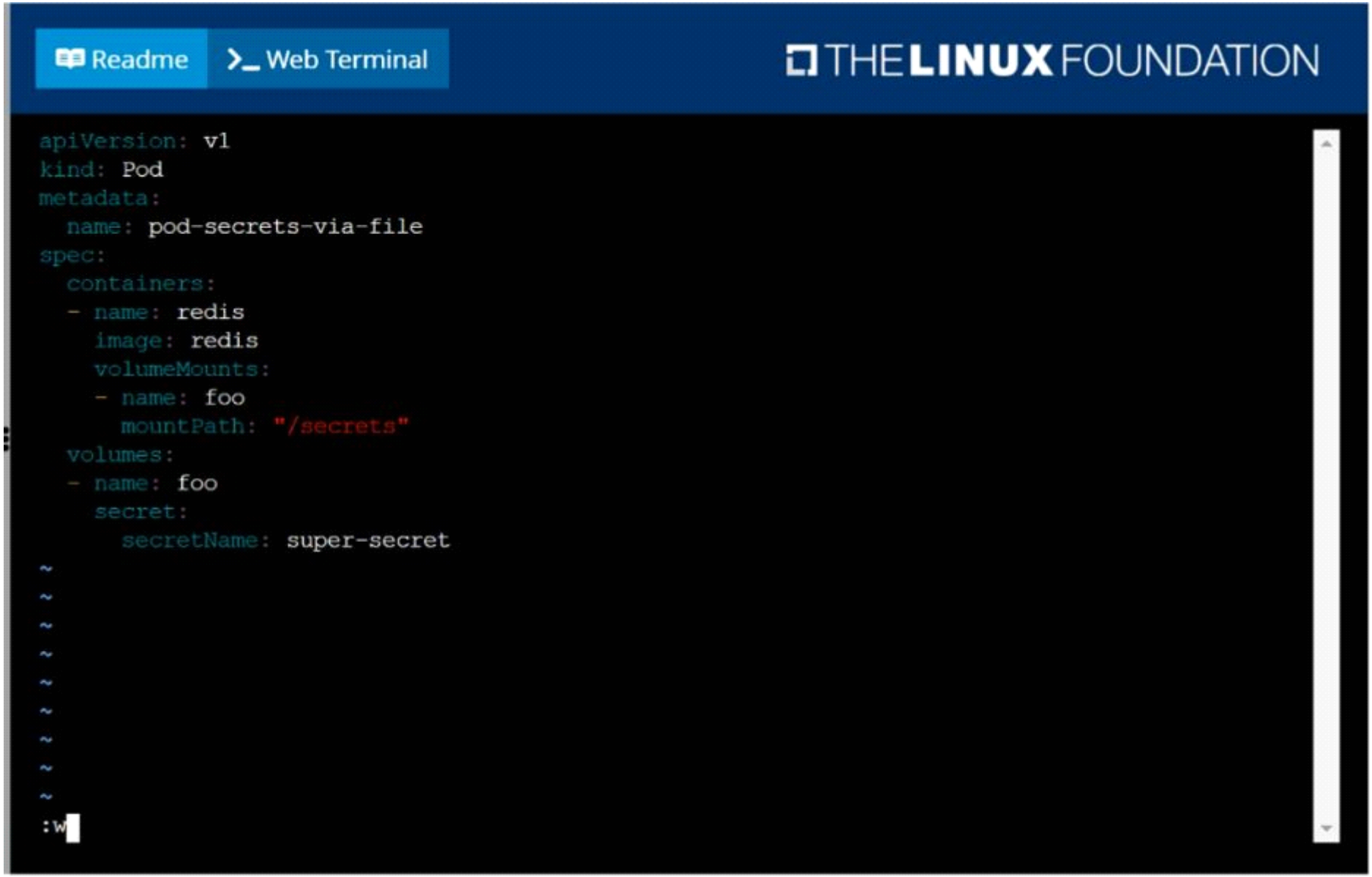
Create a pod named pod-secrets-via-file, using the redis Image, which mounts a secret named super-secret at /secrets.

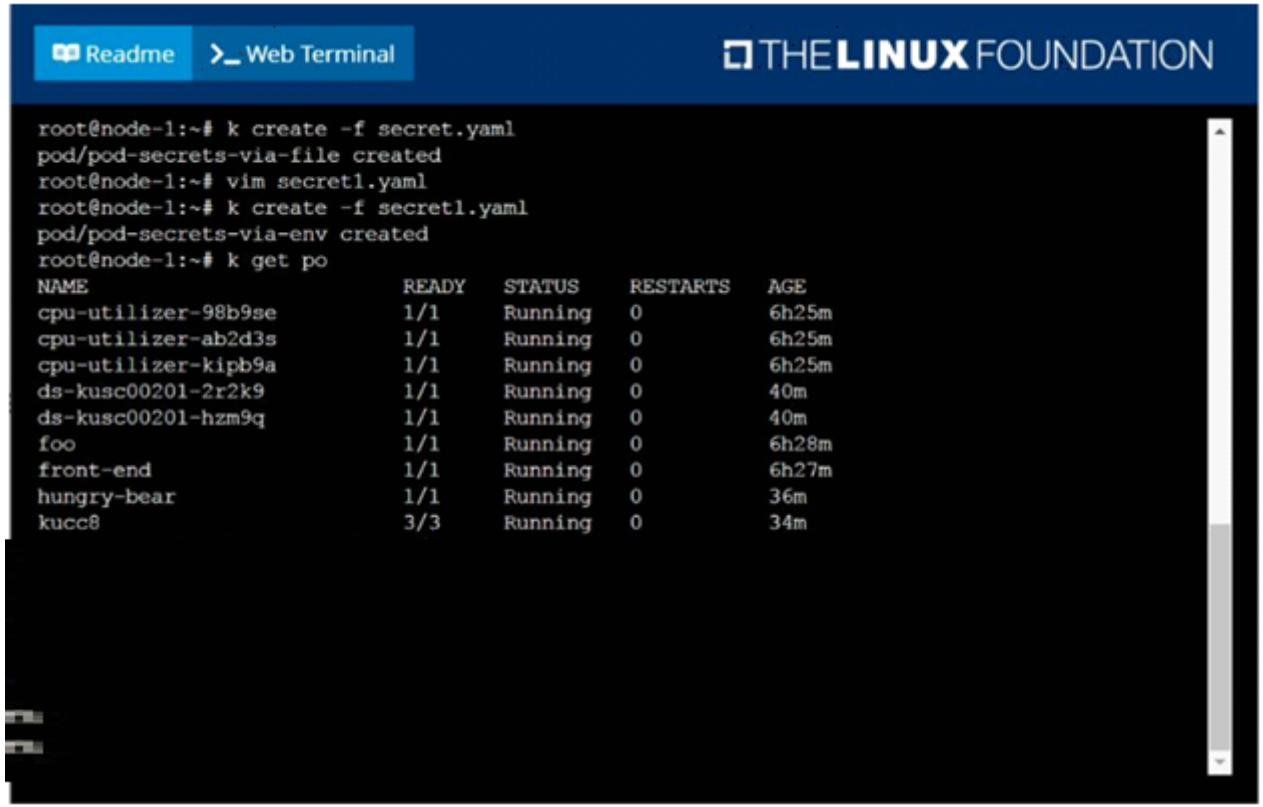
Create a second pod named pod-secrets-via-env, using the redis Image, which exports password as CONFIDENTIAL

## Options:

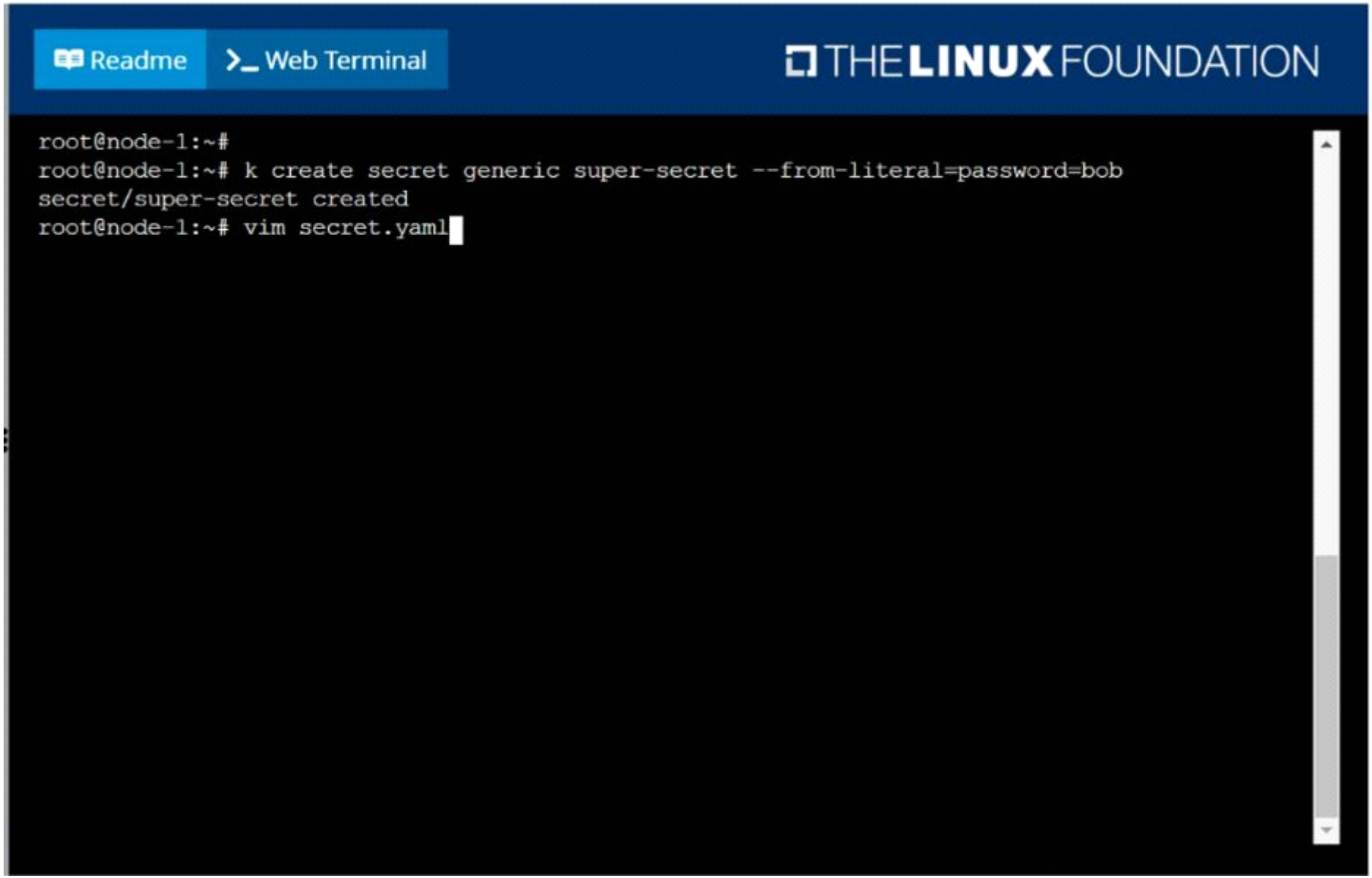
**A)** solution

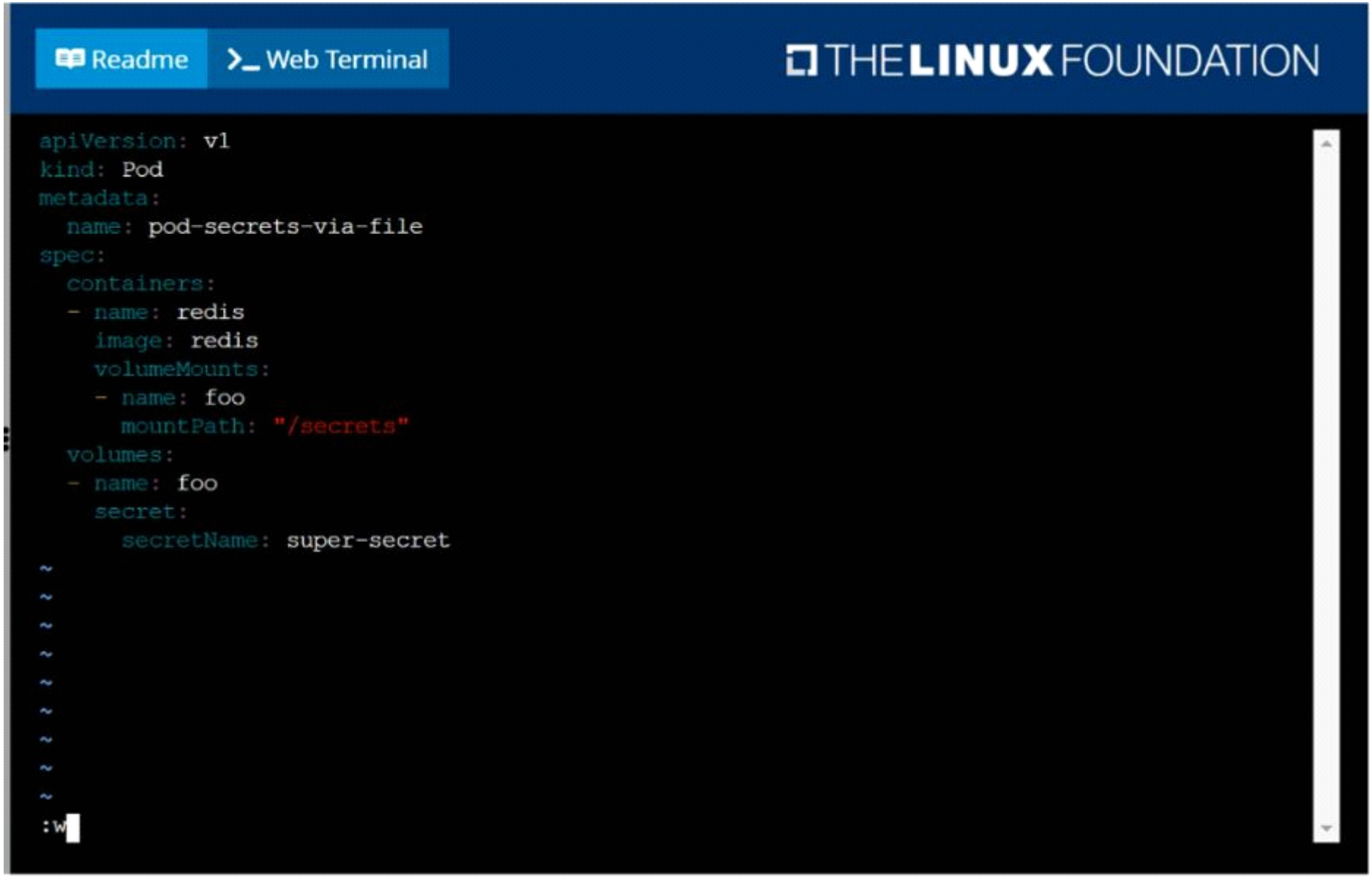

Readme
Web Terminal
THE LINUX FOUNDATION
root@node-1:~#
root@node-1:~# k create secret generic super-secret --from-literal=password=bob
secret/super-secret created
root@node-1:~# vim secret.yaml

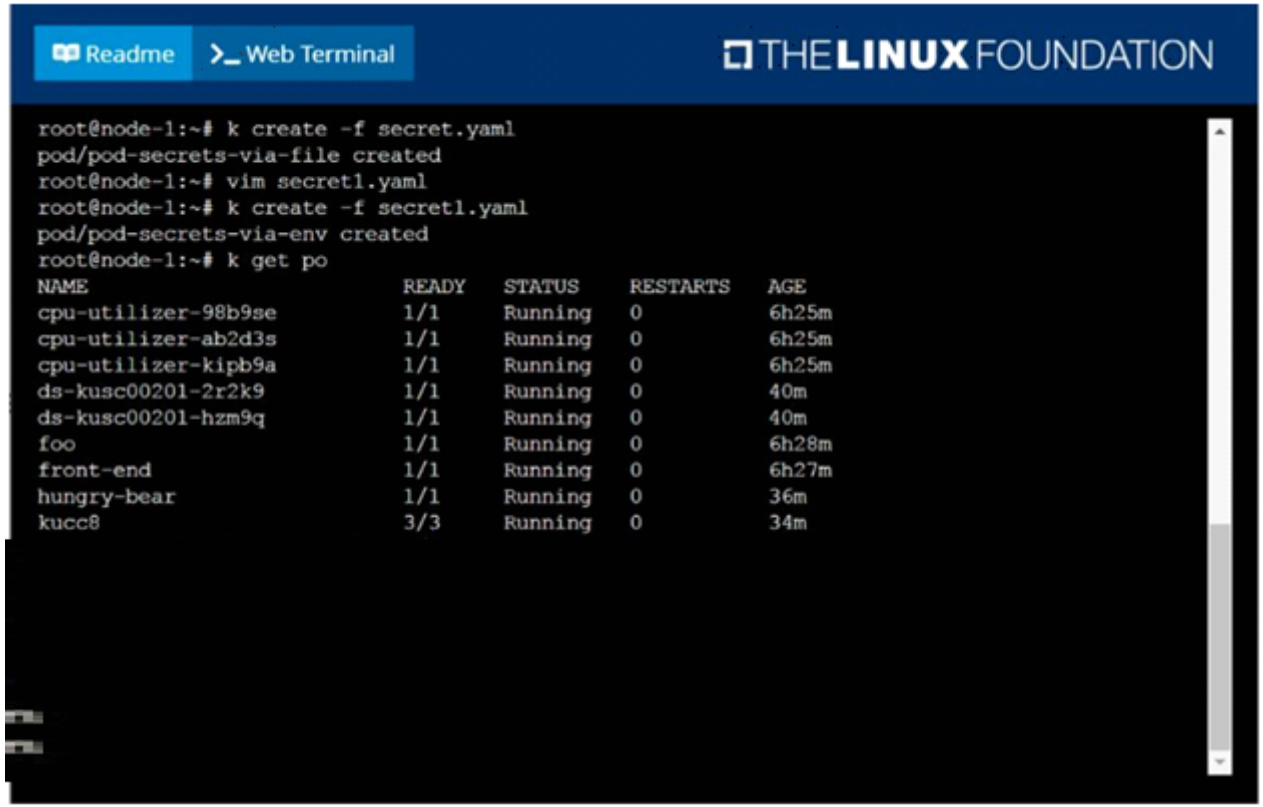

Readme
Web Terminal
THE LINUX FOUNDATION
apiVersion: v1
kind: Pod
metadata:
name: pod-secrets-via-file
spec:
containers:
- name: redis
image: redis
volumeMounts:
- name: foo
mountPath: "/secrets"
volumes:
- name: foo
secret:
secretName: super-secret
~
~
~
~
~
~
~
~
~
~
:w


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

**B)** solution


Readme
Web Terminal
THE LINUX FOUNDATION
root@node-1:~#
root@node-1:~# k create secret generic super-secret --from-literal=password=bob
secret/super-secret created
root@node-1:~# vim secret.yaml


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




### Answer:

A

## Question 3

**Question Type:** MultipleChoice

SIMULATION

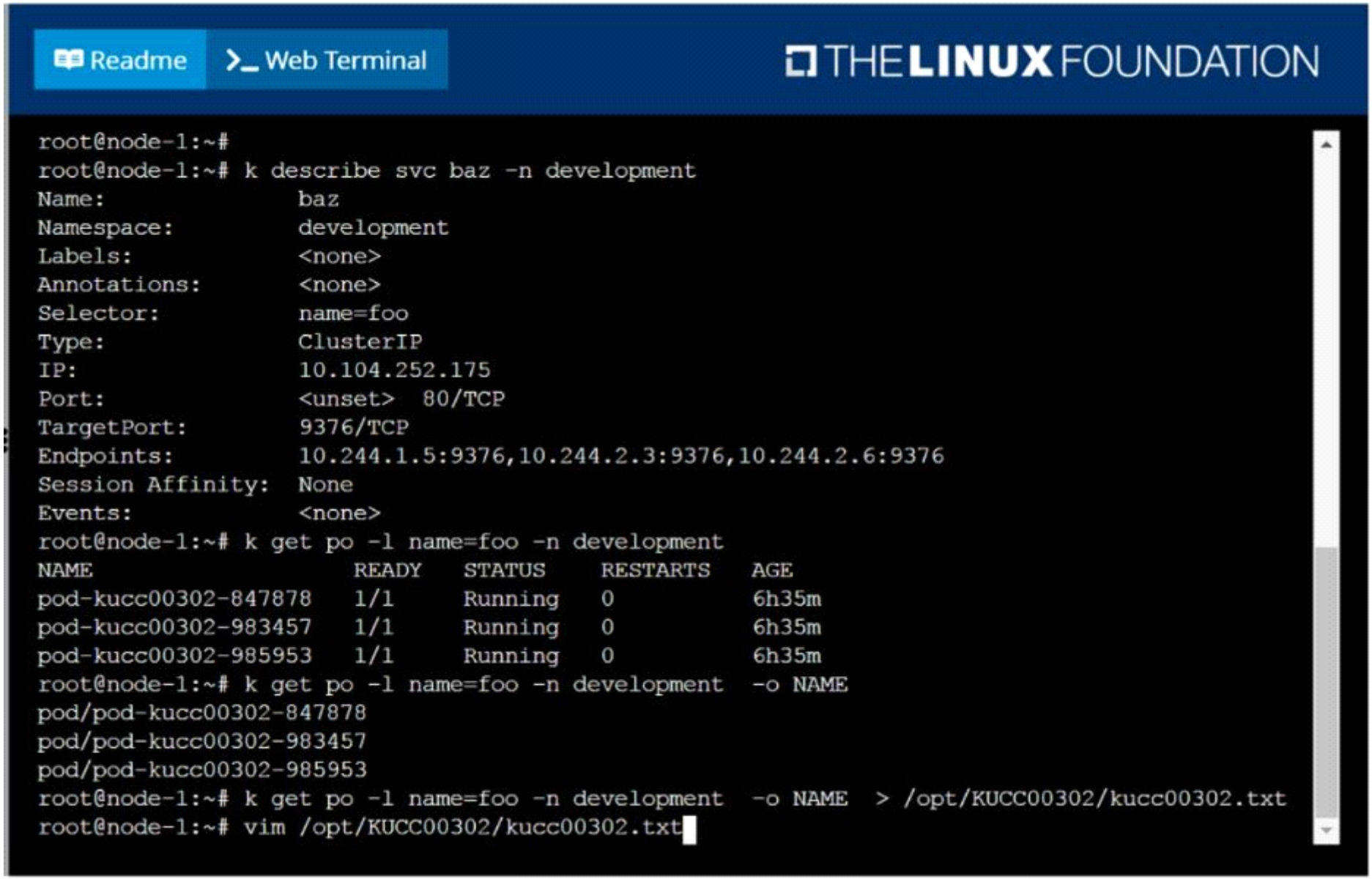
Create a file:

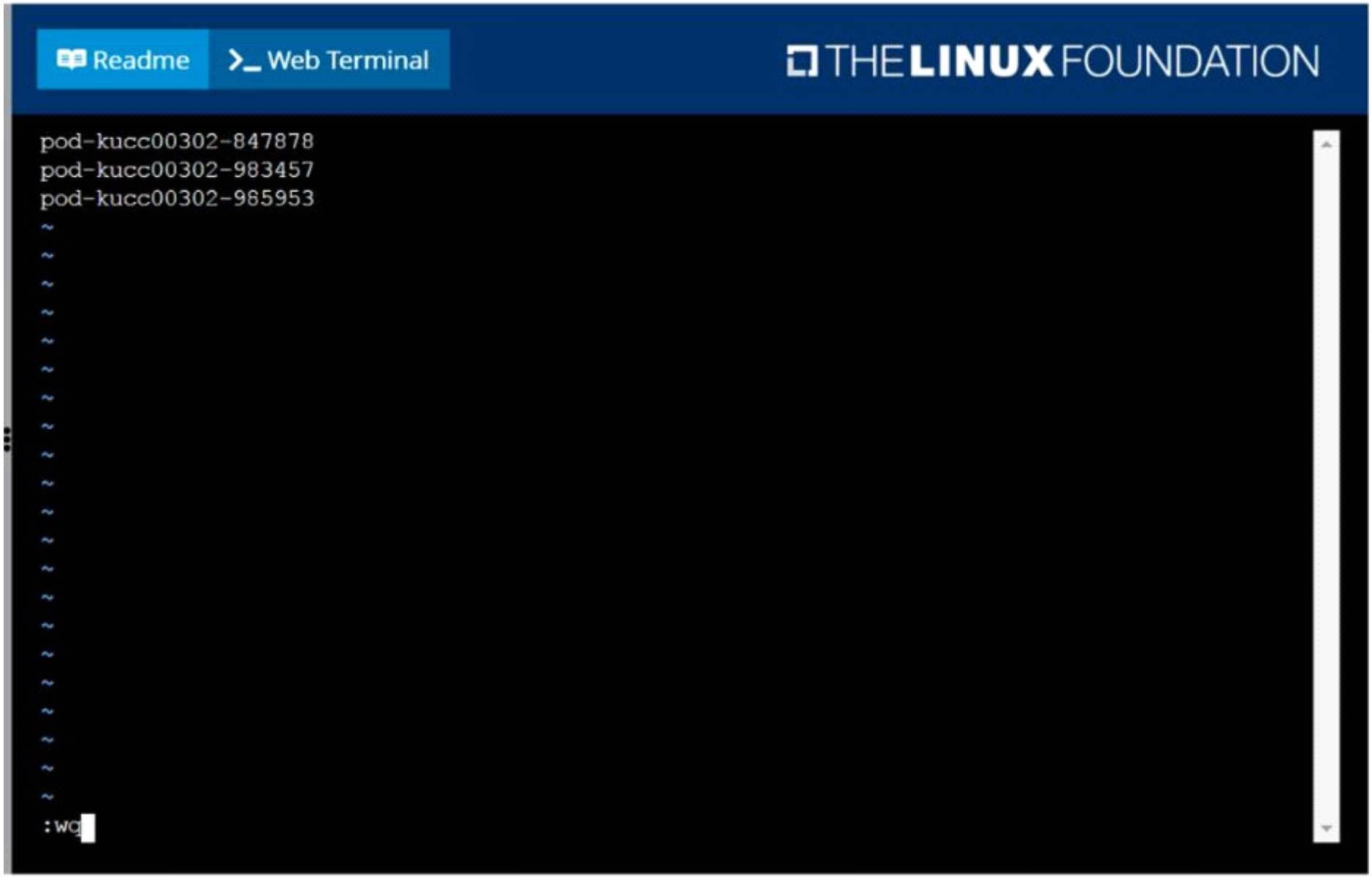
/opt/KUCC00302/kucc00302.txt that lists all pods that implement service baz in namespace development.

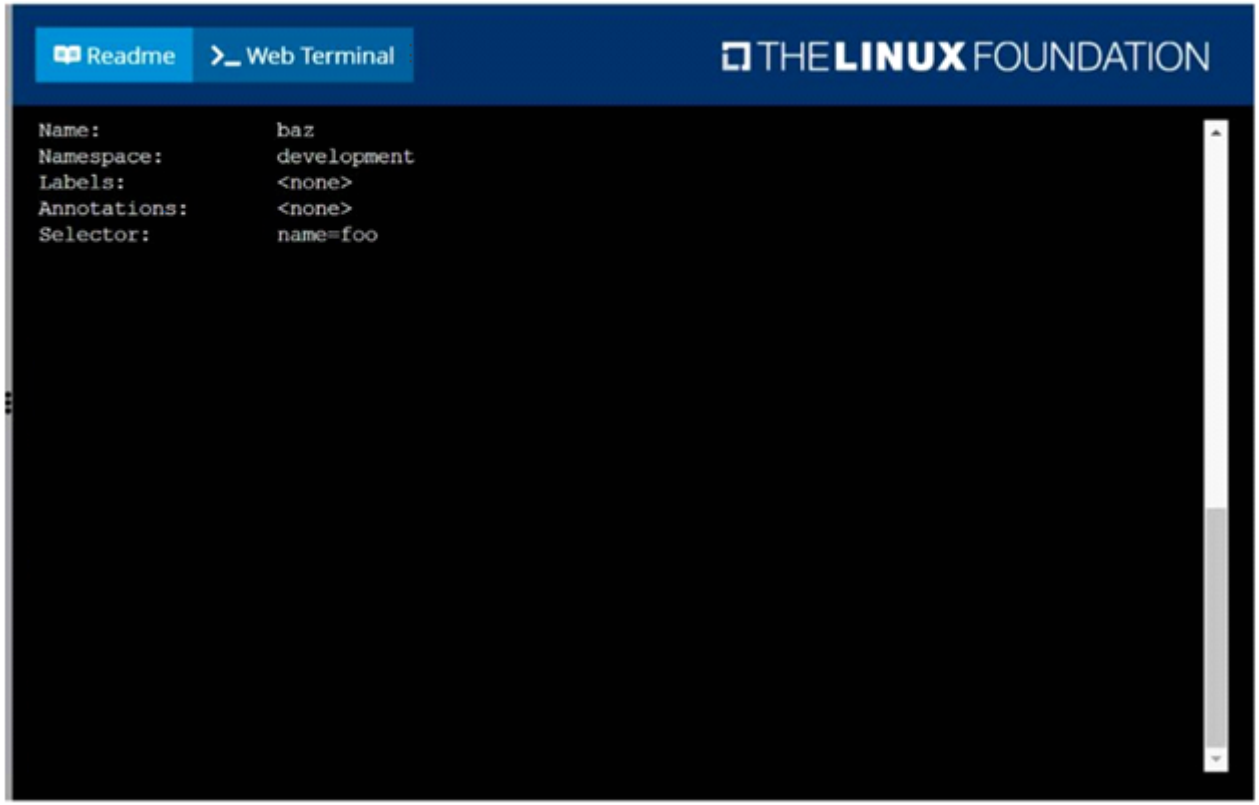
The format of the file should be one pod name per line.

## Options:

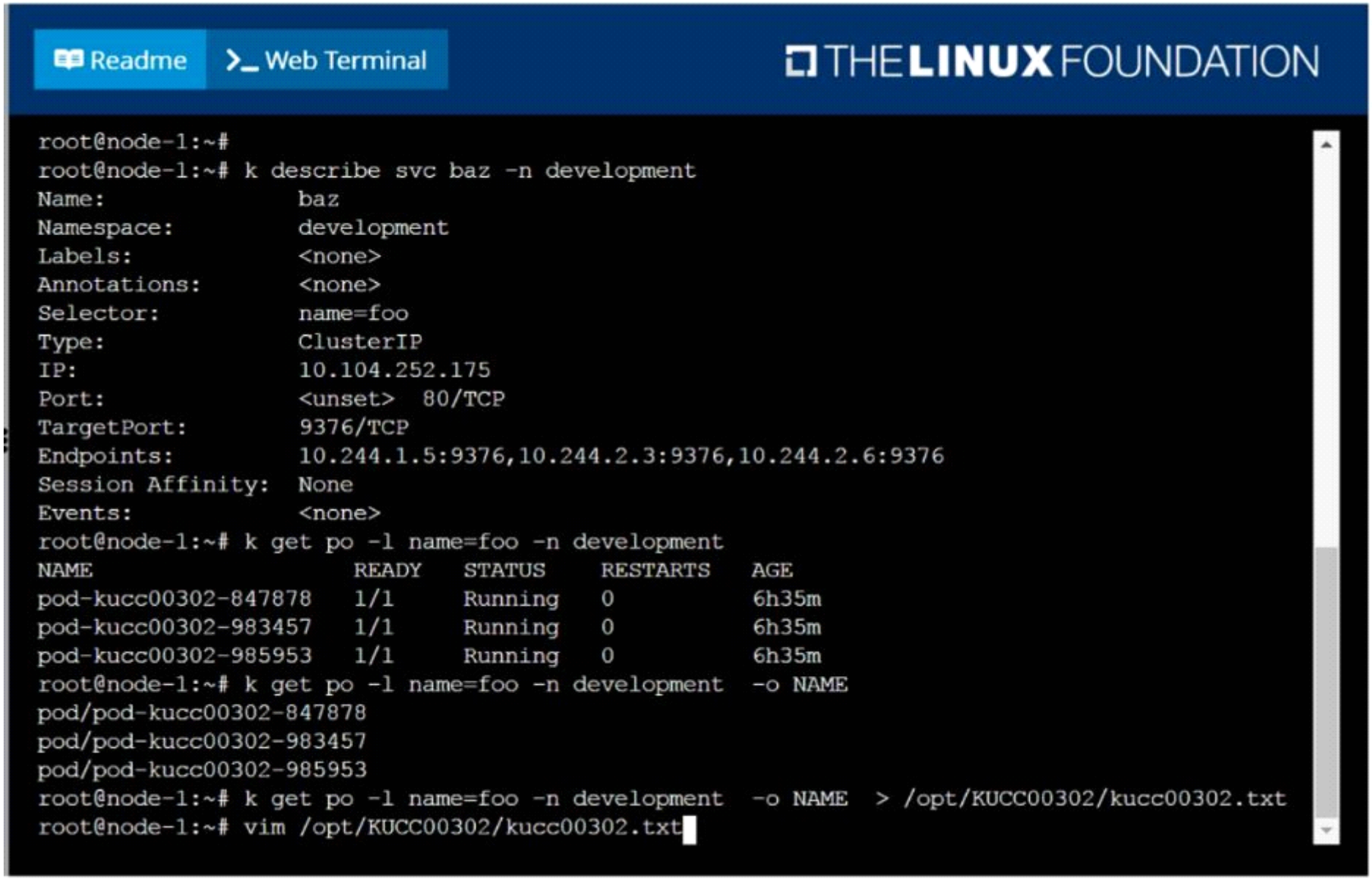
**A)** solution

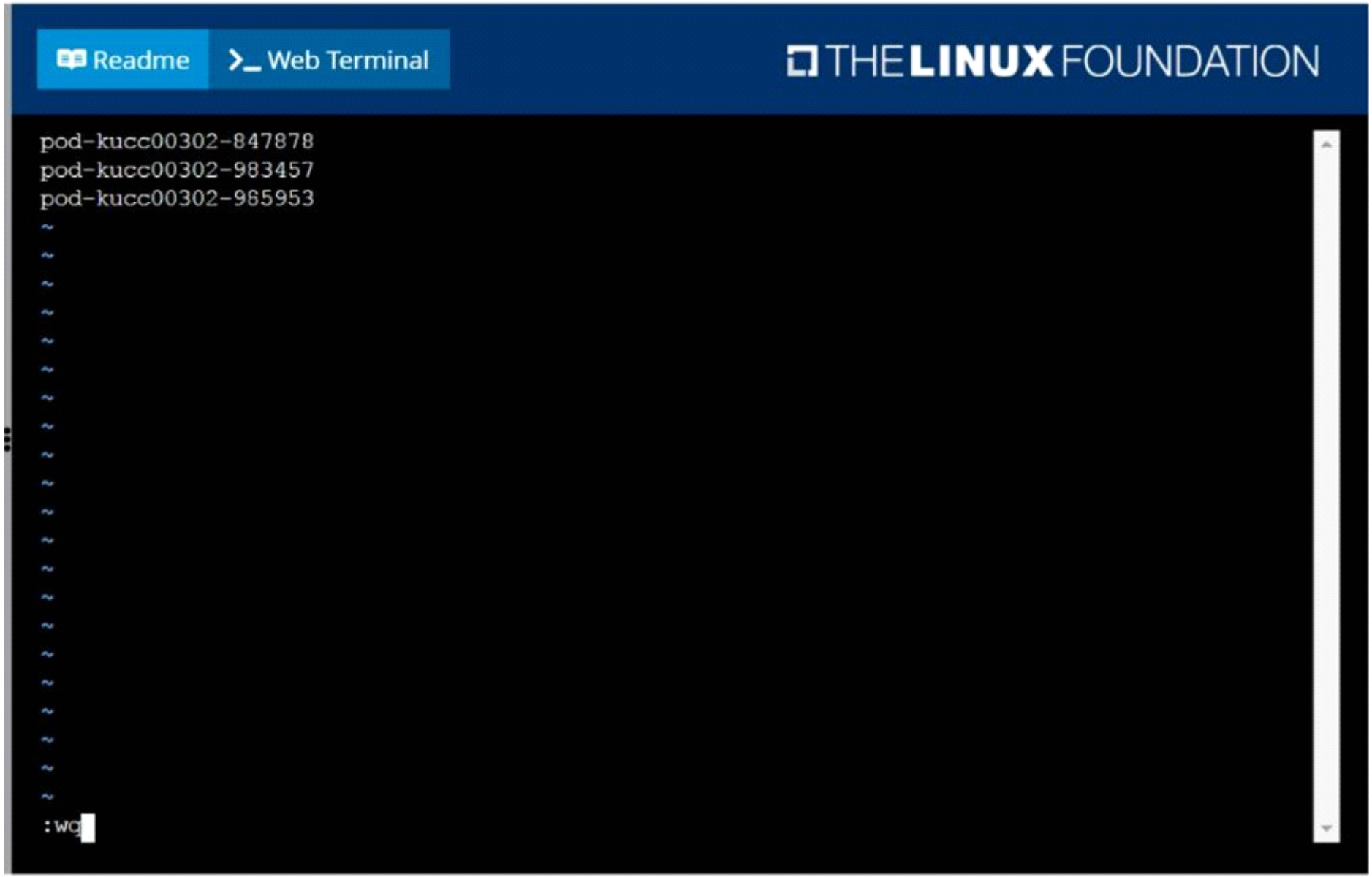

Readme
Web Terminal
THE LINUX FOUNDATION
root@node-1:~#
root@node-1:~# k describe svc baz -n development
Name: baz
Namespace: development
Labels: <none>
Annotations: <none>
Selector: name=foo
Type: ClusterIP
IP: 10.104.252.175
Port: <unset> 80/TCP
TargetPort: 9376/TCP
Endpoints: 10.244.1.5:9376,10.244.2.3:9376,10.244.2.6:9376
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

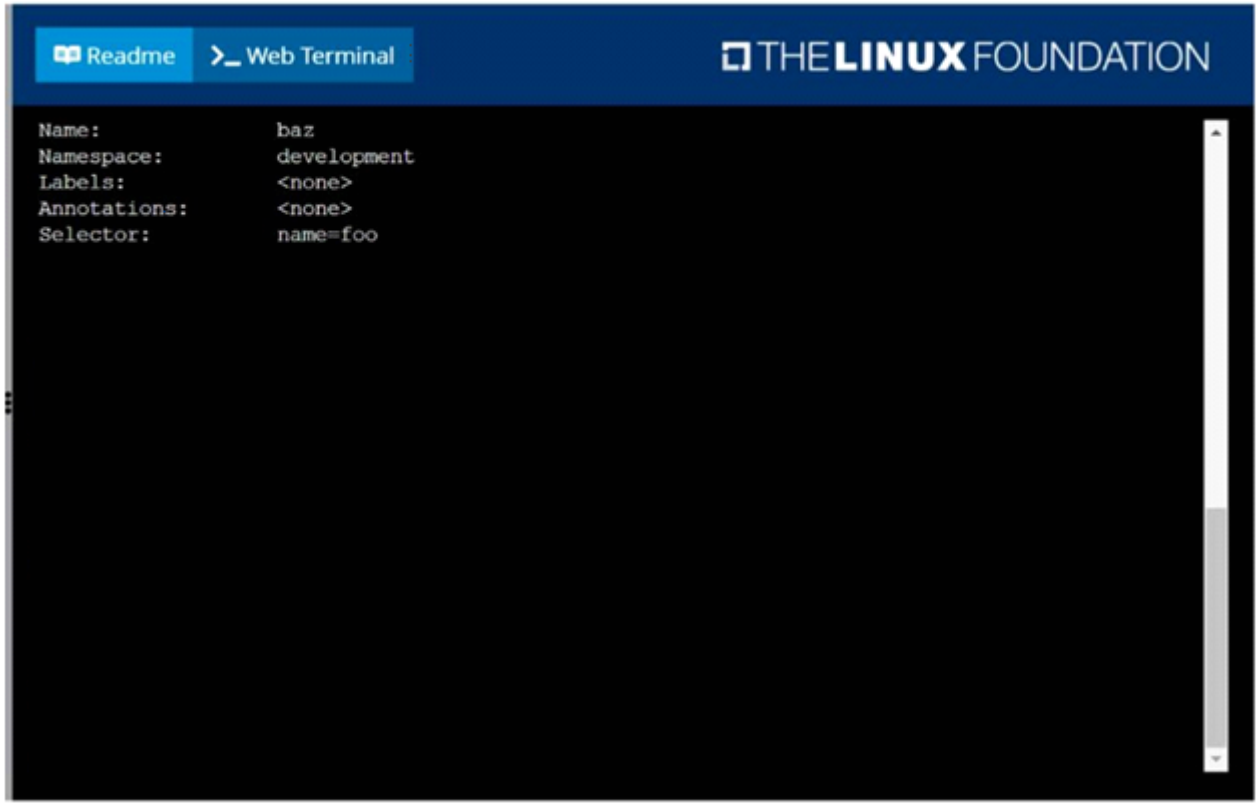

Readme
Web Terminal
THE LINUX FOUNDATION
pod-kucc00302-847878
pod-kucc00302-983457
pod-kucc00302-965953
:wq


Readme
Web Terminal
THE LINUX FOUNDATION
Name: baz
Namespace: development
Labels: <none>
Annotations: <none>
Selector: name=foo

**B)** solution


Readme
Web Terminal
THE LINUX FOUNDATION
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


Readme
Web Terminal
THE LINUX FOUNDATION
pod-kucc00302-847878
pod-kucc00302-983457
pod-kucc00302-965953
:wq




### Answer:

A

## Question 4

**Question Type:** MultipleChoice

SIMULATION

Create a deployment spec file that will:

* Launch 7 replicas of the nginx Image with the label

app_runtime_stage=dev

* deployment name: kual00201

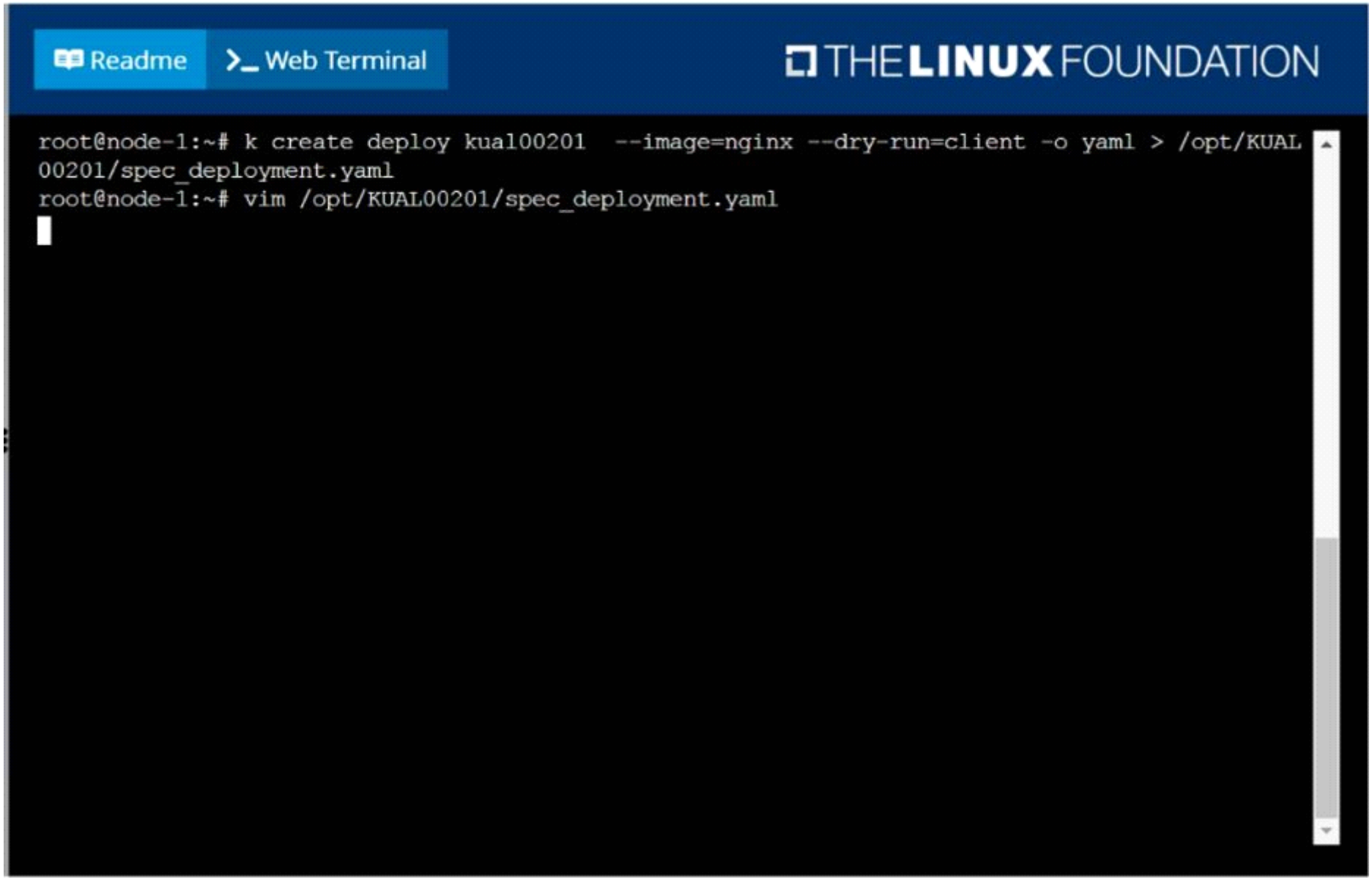
Save a copy of this spec file to /opt/KUAL00201/spec_deployment.yaml

(or /opt/KUAL00201/spec_deployment.json).

When you are done, clean up (delete) any new Kubernetes API object that you produced during this task.

## Options:

**A)** solution

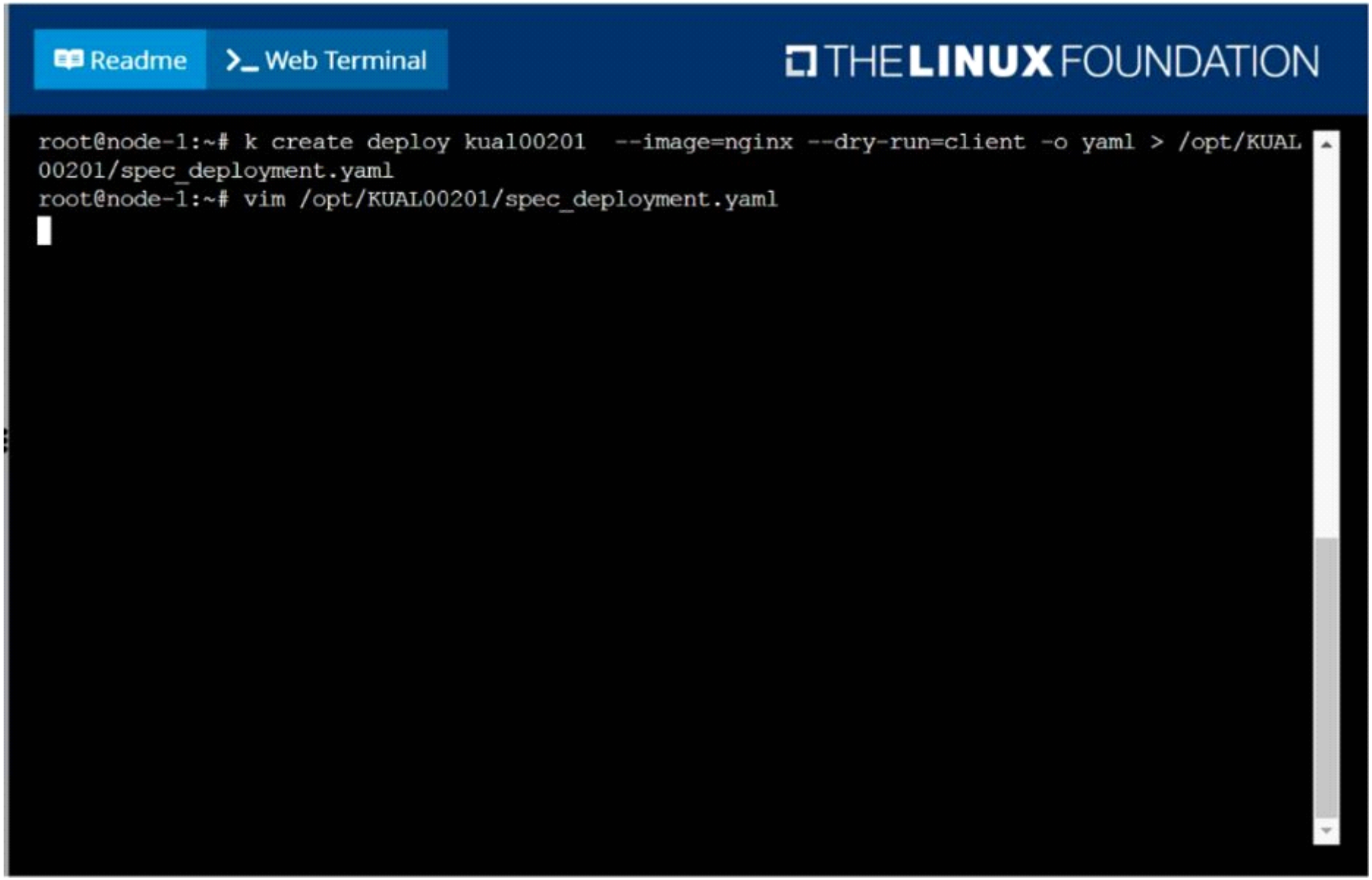

Readme
Web Terminal
THE LINUX FOUNDATION
root@node-1:~# k create deploy kual00201 --image=nginx --dry-run=client -o yaml > /opt/KUAL
00201/spec_deployment.yaml
root@node-1:~# vim /opt/KUAL00201/spec_deployment.yaml

Readme | Web Terminal

THE LINUX FOUNDATION

```
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
```

**B)** solution


Readme
Web Terminal
THE LINUX FOUNDATION
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

Readme | Web Terminal

THE LINUX FOUNDATION

```
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
```

## Answer:

B

# Question 5

**Question Type:** MultipleChoice

SIMULATION

Create a pod as follows:

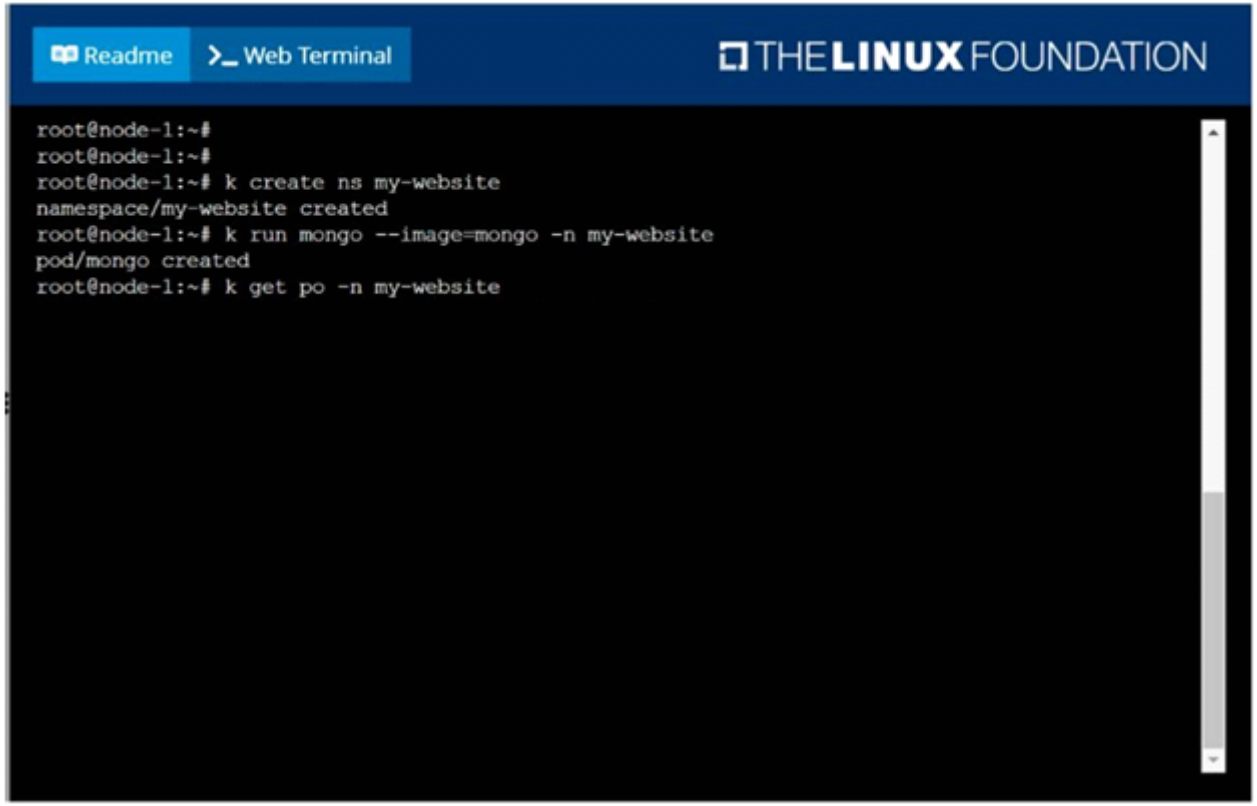
* Name: mongo

* Using Image: mongo

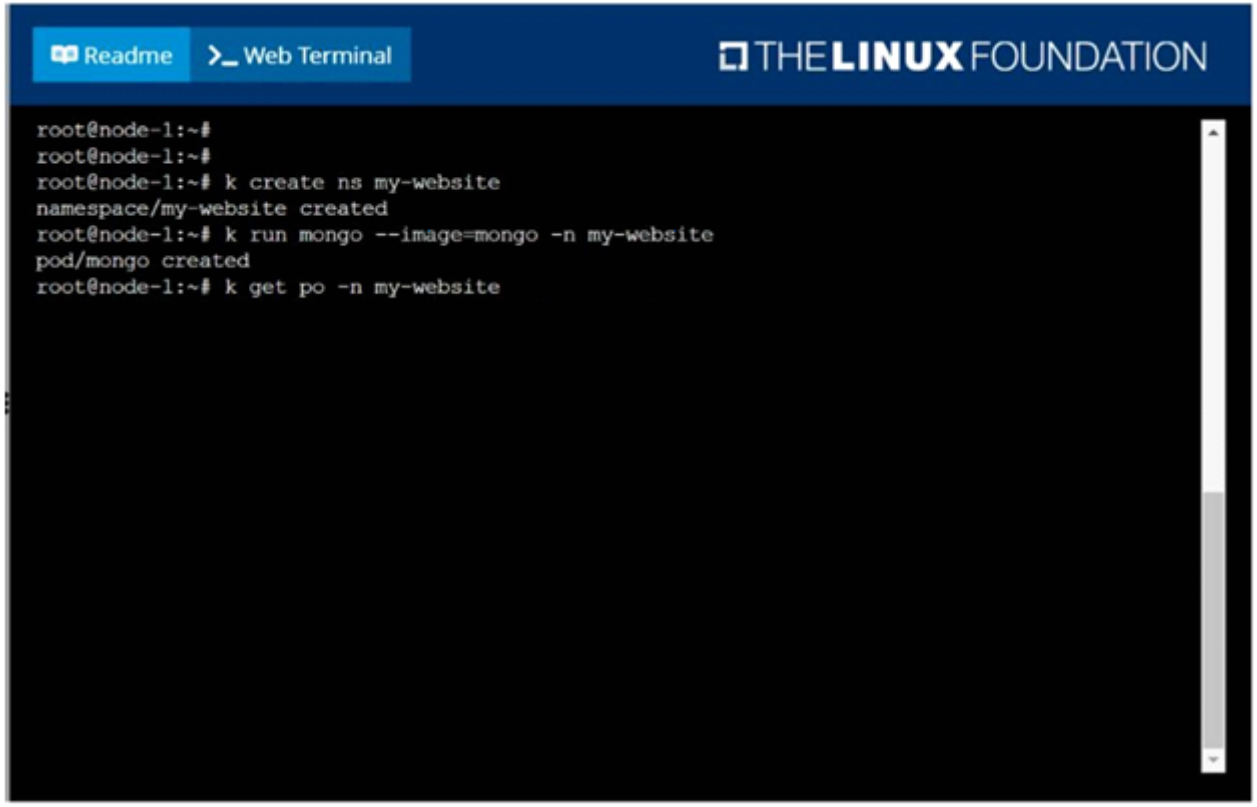
* In a new Kubernetes namespace named: my-website

## Options:

**A)** 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

**B)** solution




### Answer:

A

## Question 6

**Question Type:** MultipleChoice

SIMULATION

Create a pod as follows:

* Name: non-persistent-redis

* container Image: redis

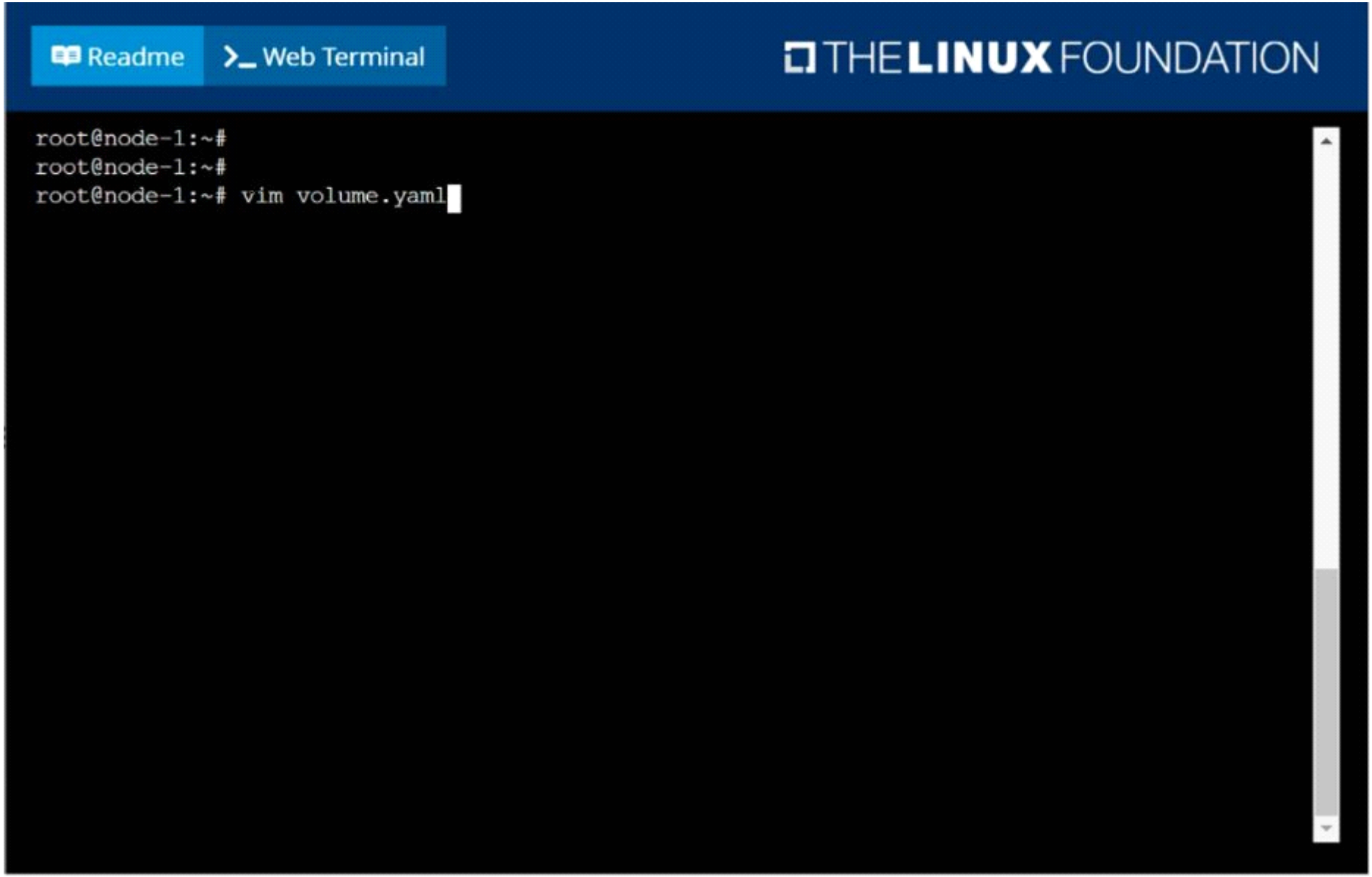
* Volume with name: cache-control

* Mount path: /data/redis

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

## Options:

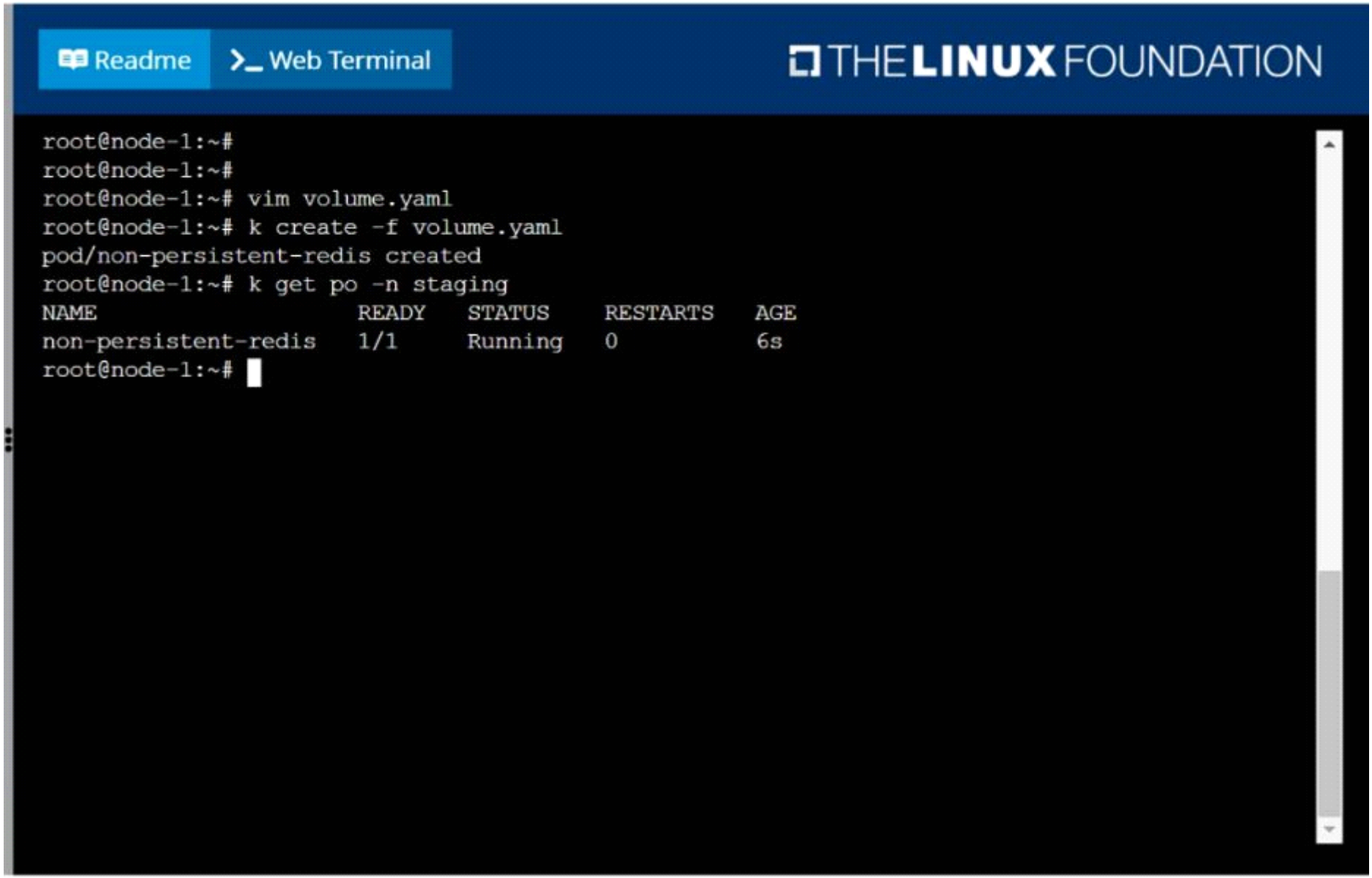
**A)** solution


Readme
Web Terminal
THE LINUX FOUNDATION
root@node-1:~#
root@node-1:~#
root@node-1:~# vim volume.yaml

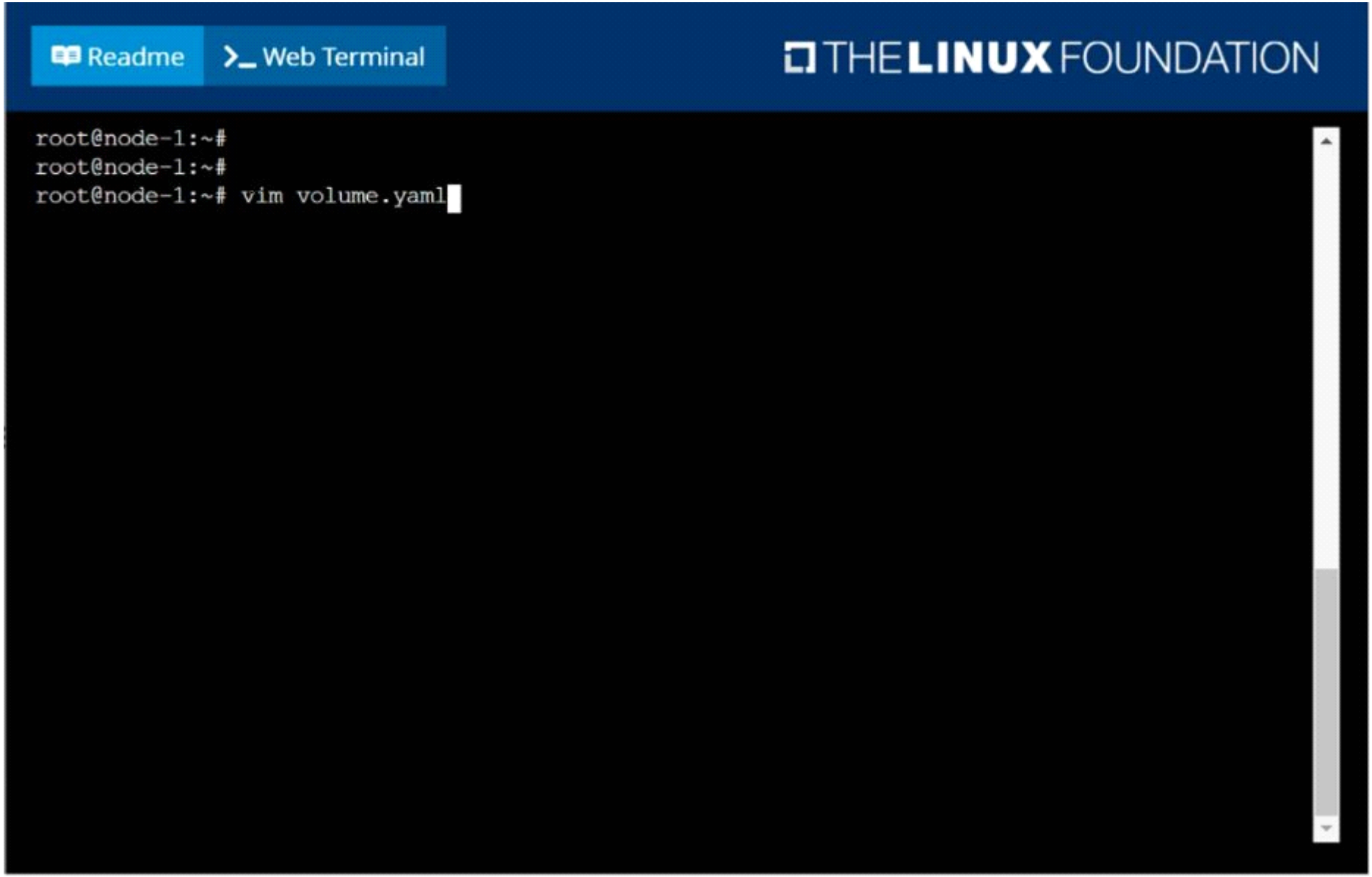
Readme | Web Terminal

THE LINUX FOUNDATION

```
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
```


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:~#

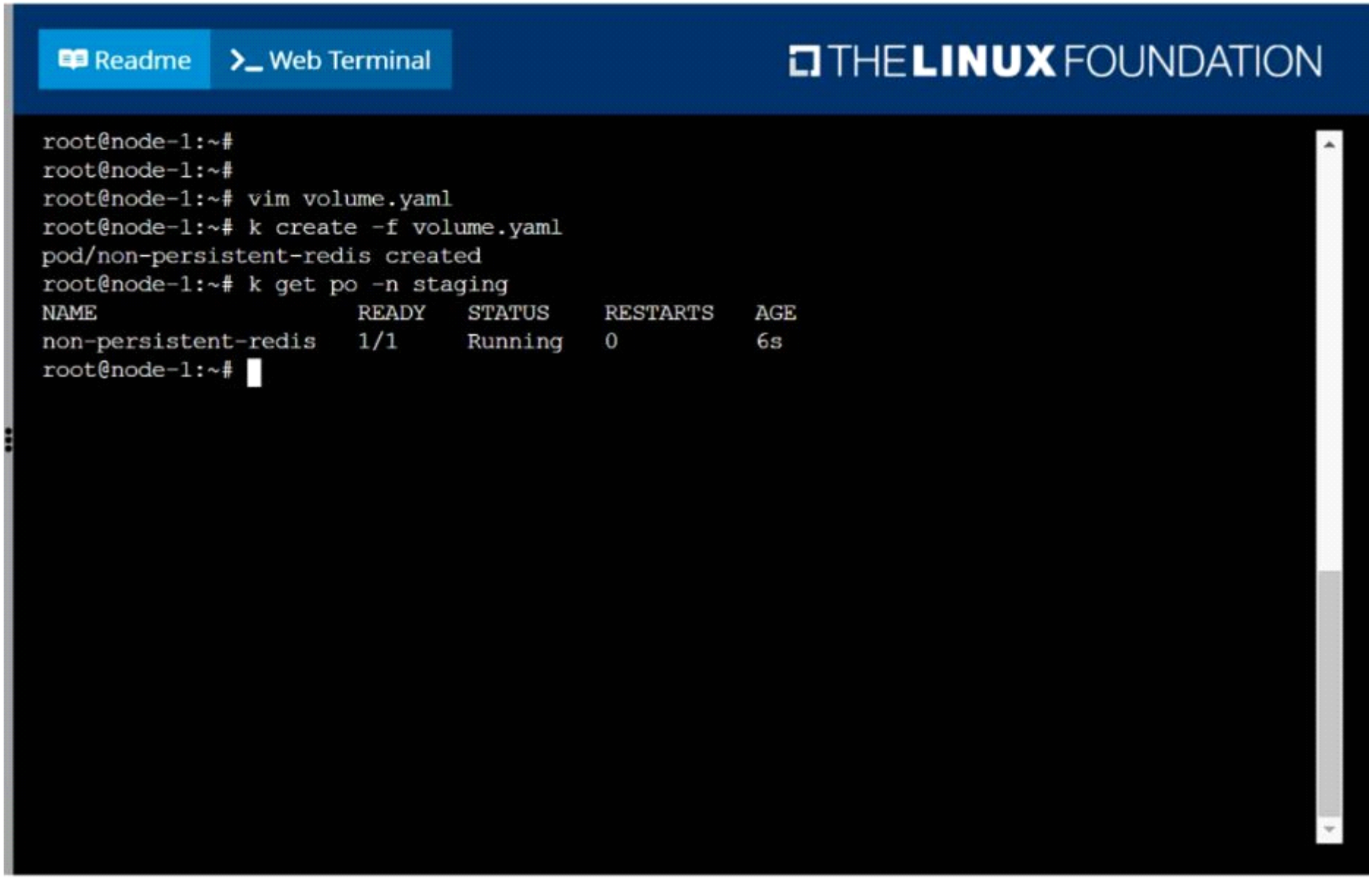
**B)** solution


Readme
Web Terminal
THE LINUX FOUNDATION
root@node-1:~#
root@node-1:~#
root@node-1:~# vim volume.yaml

Readme | Web Terminal

THE LINUX FOUNDATION

```
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
```


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:~#

## Answer:

B

# Question 7

**Question Type:** MultipleChoice

SIMULATION

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:

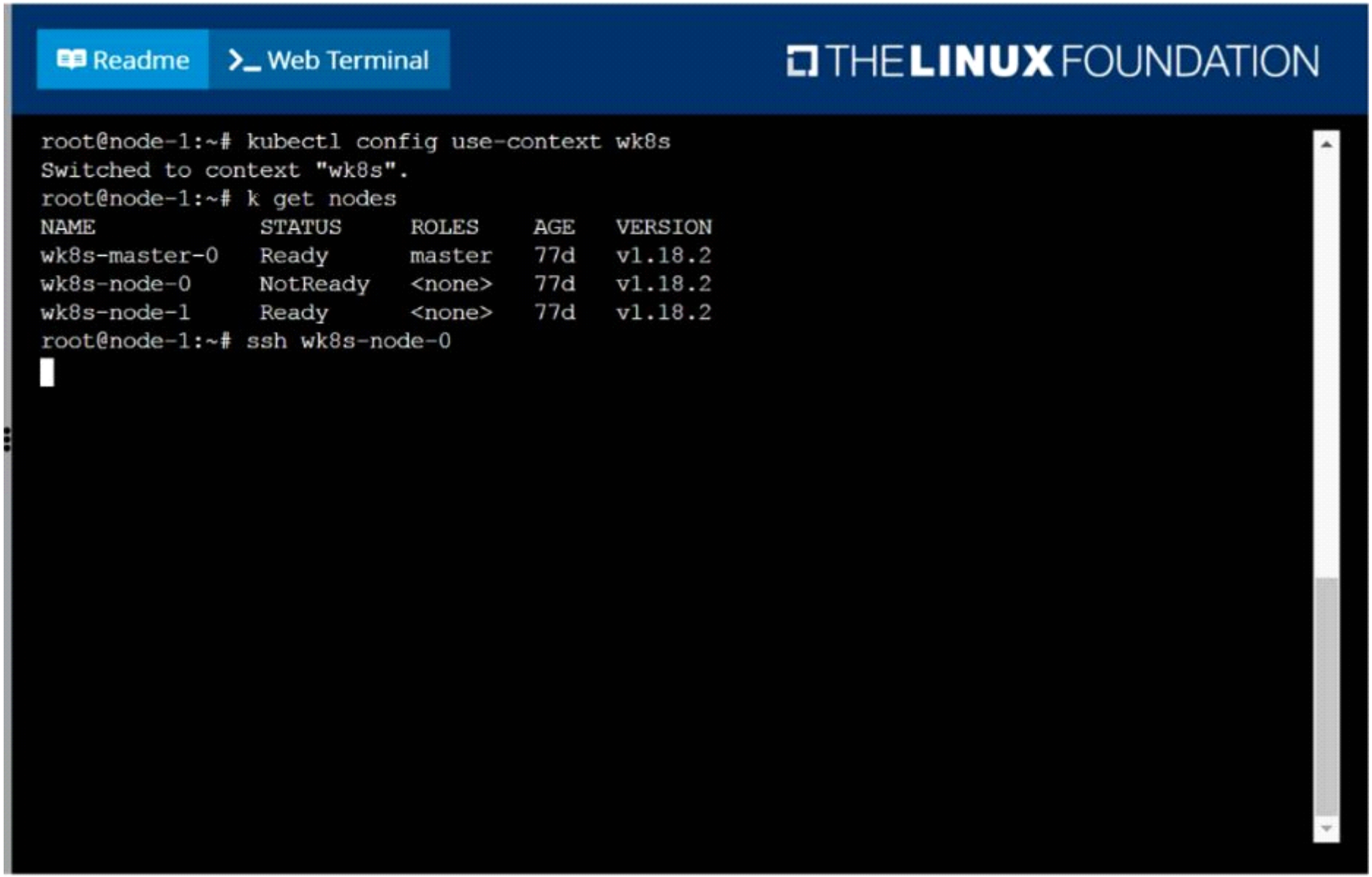
[student@node-1] $ | ssh Wk8s-node-0

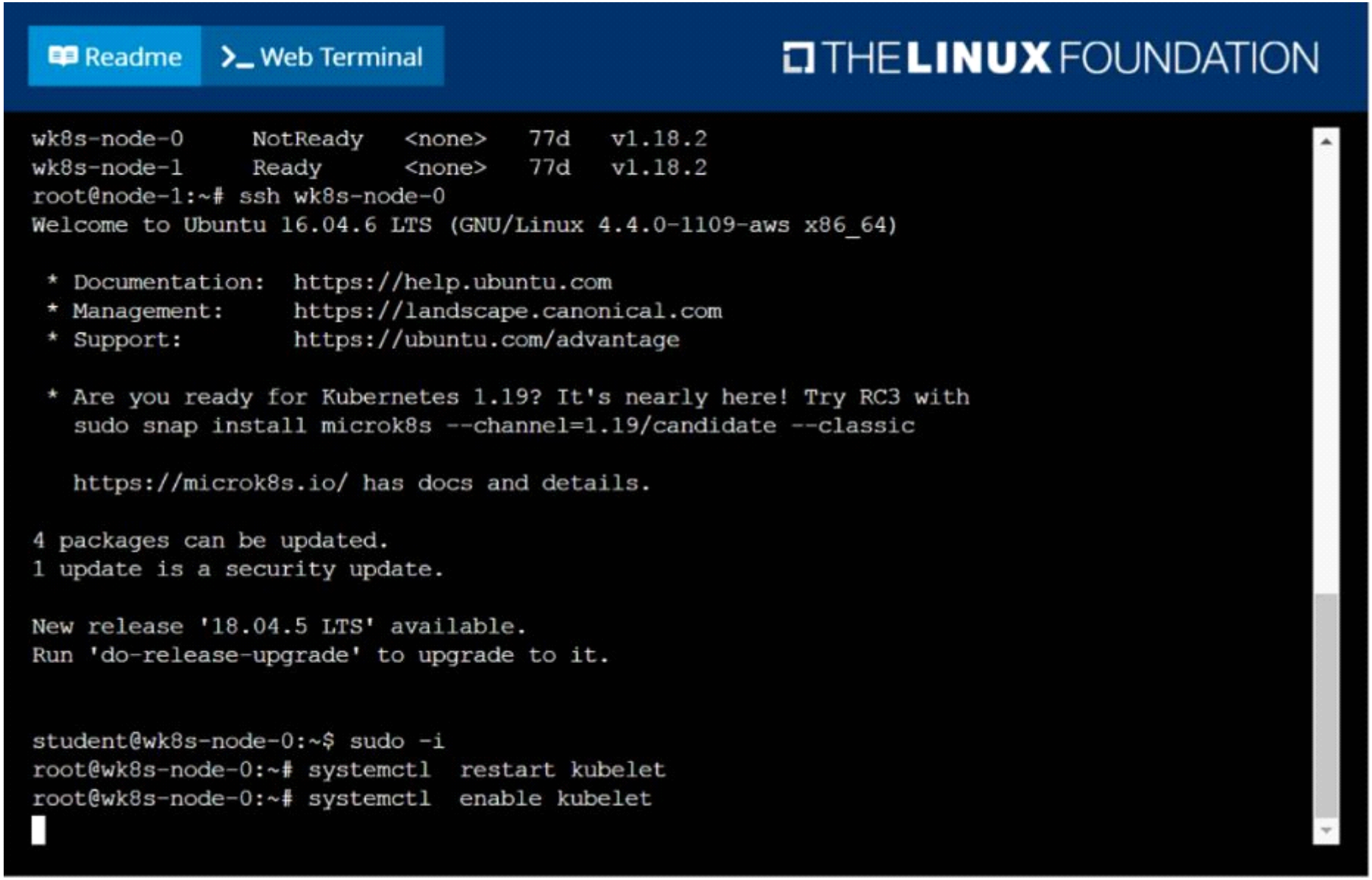
You can assume elevated privileges on the node with the following command:

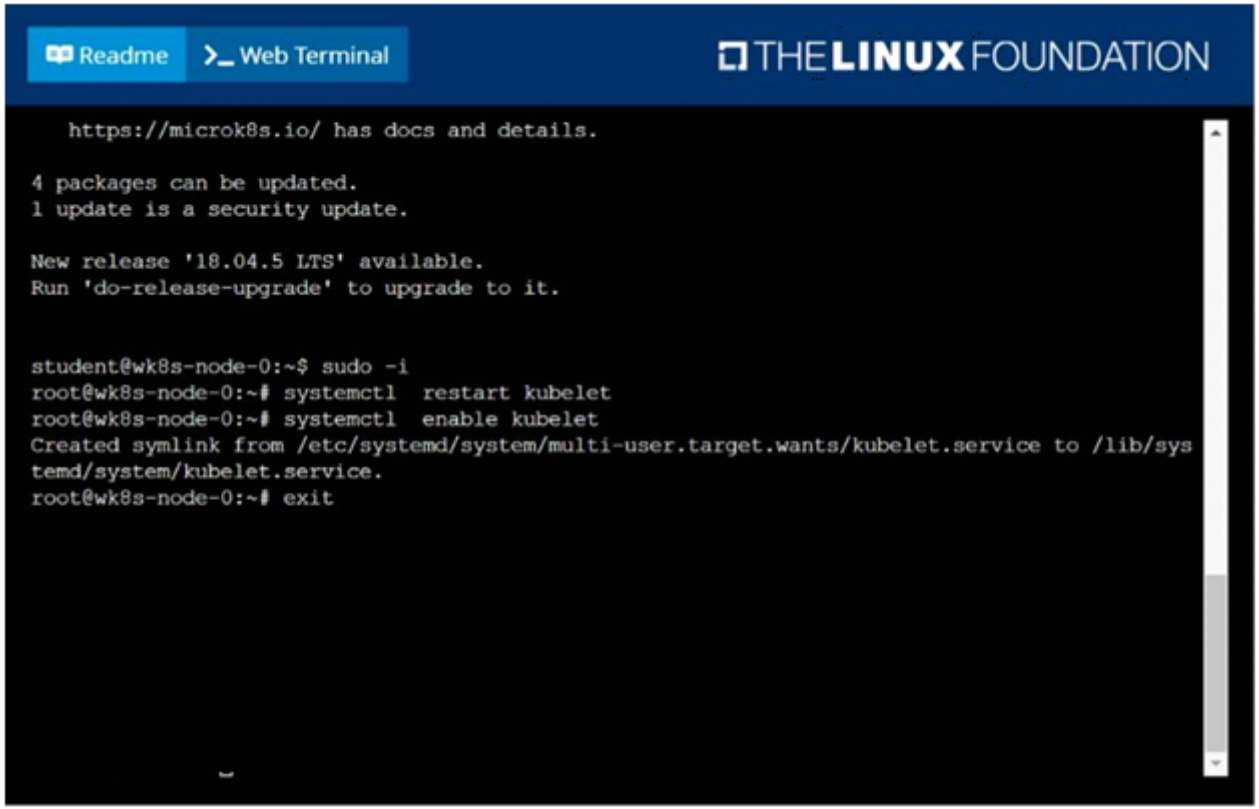
[student@w8ks-node-0] $ | sudo --i

## Options:

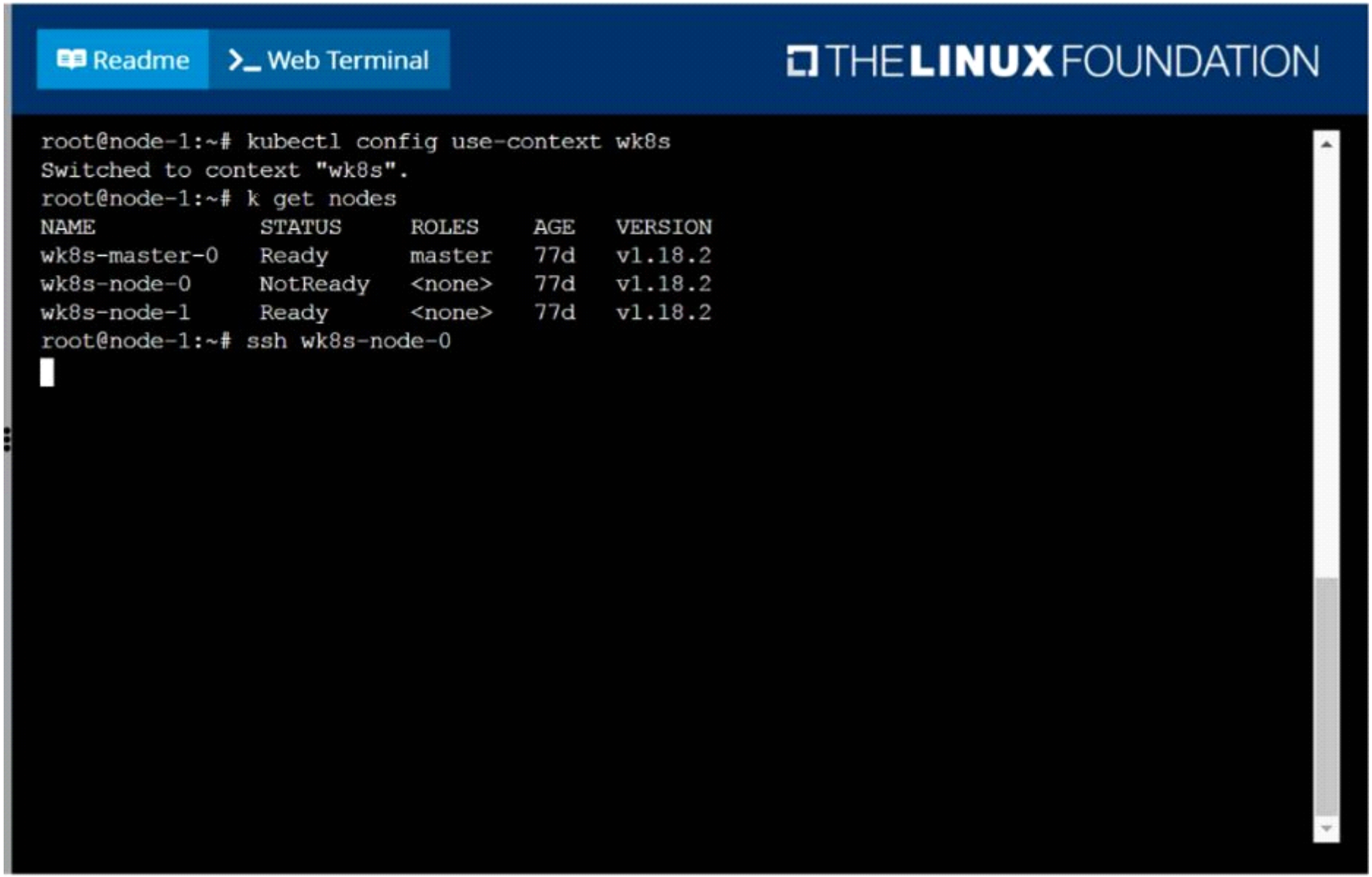
**A)** solution

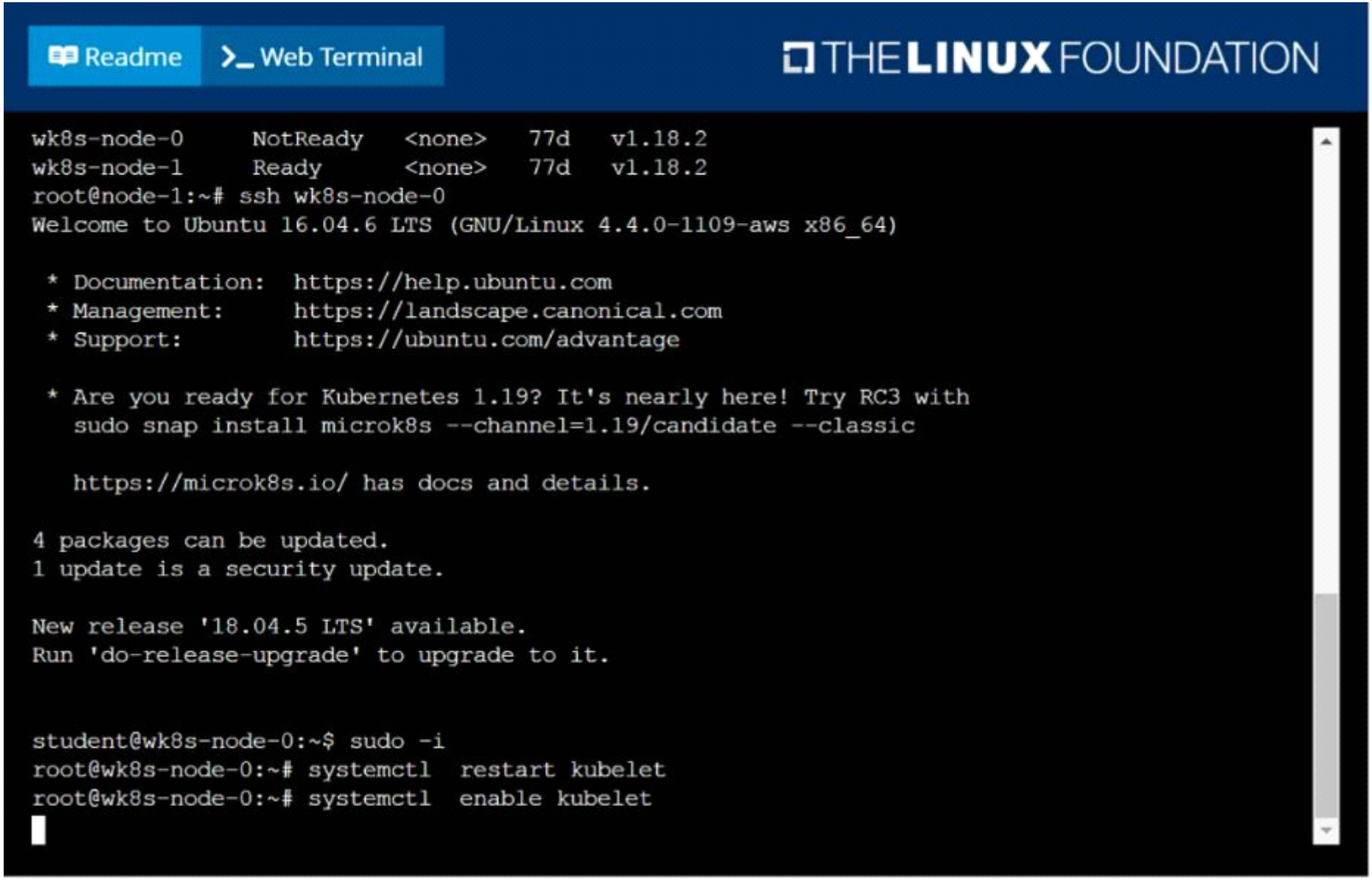

Readme
Web Terminal
THE LINUX FOUNDATION
root@node-1:~# kubectl config use-context wk8s
Switched to context "wk8s".
root@node-1:~# k get nodes
NAME STATUS ROLES AGE VERSION
wk8s-master-0 Ready master 77d v1.18.2
wk8s-node-0 NotReady <none> 77d v1.18.2
wk8s-node-1 Ready <none> 77d v1.18.2
root@node-1:~#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

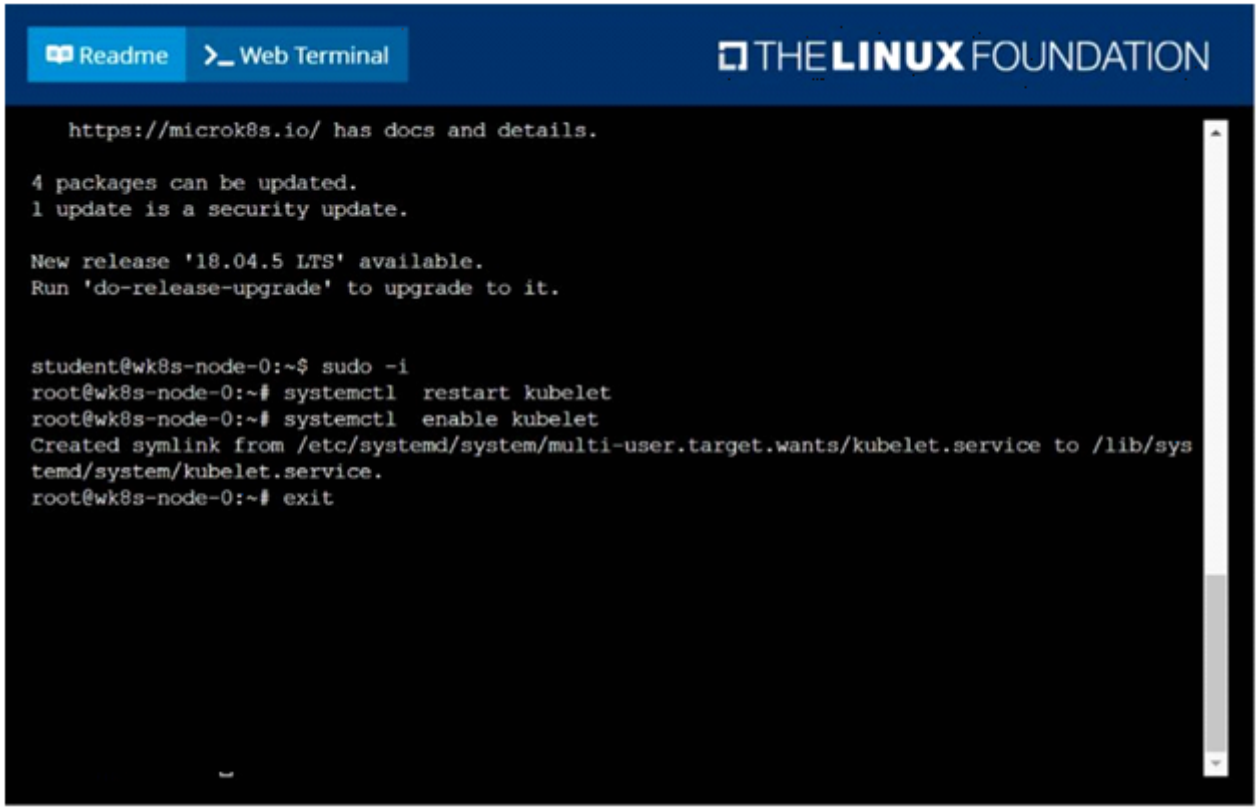

Readme
Web Terminal
THE LINUX FOUNDATION
wk8s-node-0 NotReady <none> 77d v1.18.2
wk8s-node-1 Ready <none> 77d v1.18.2
root@node-1:~# ssh wk8s-node-0
Welcome to Ubuntu 16.04.6 LTS (GNU/Linux 4.4.0-1109-aws x86_64)
* Documentation: https://help.ubuntu.com
* Management: https://landscape.canonical.com
* 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


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

**B)** 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


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




### Answer:

A

## Question 8

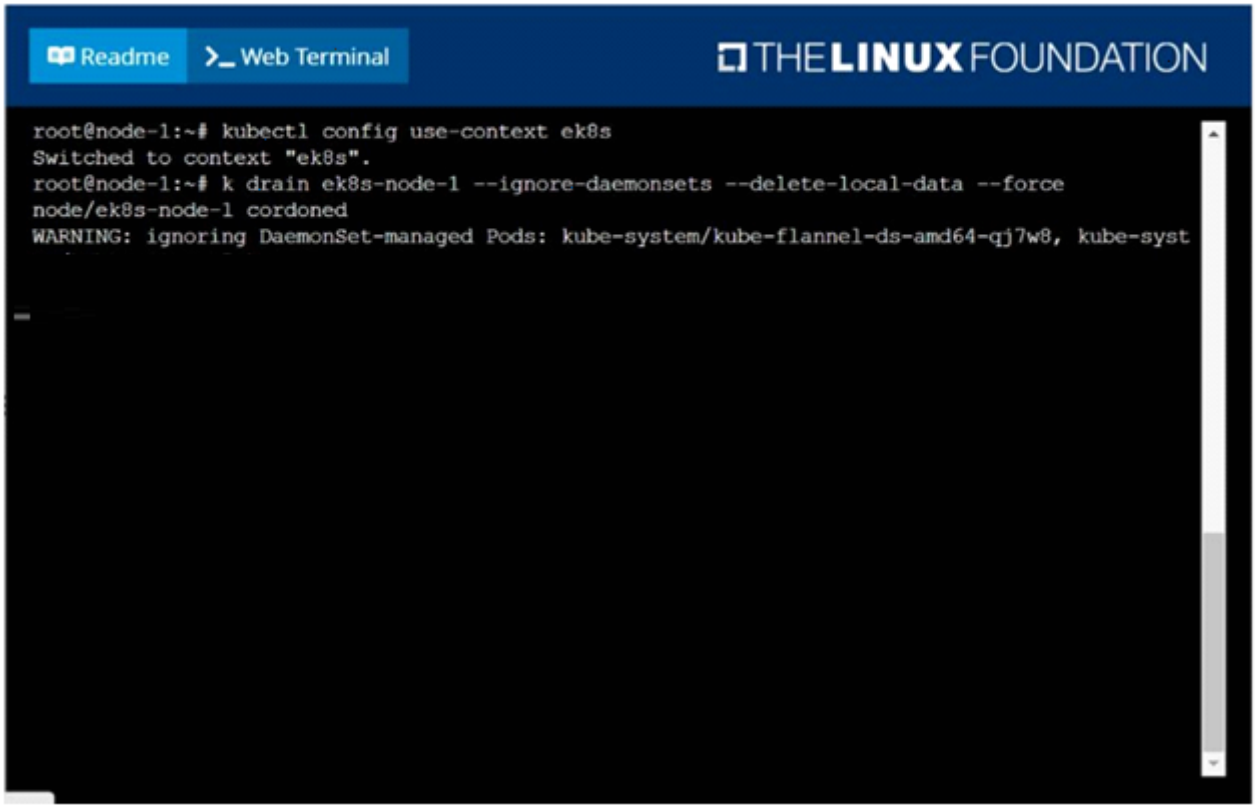
**Question Type:** MultipleChoice

SIMULATION

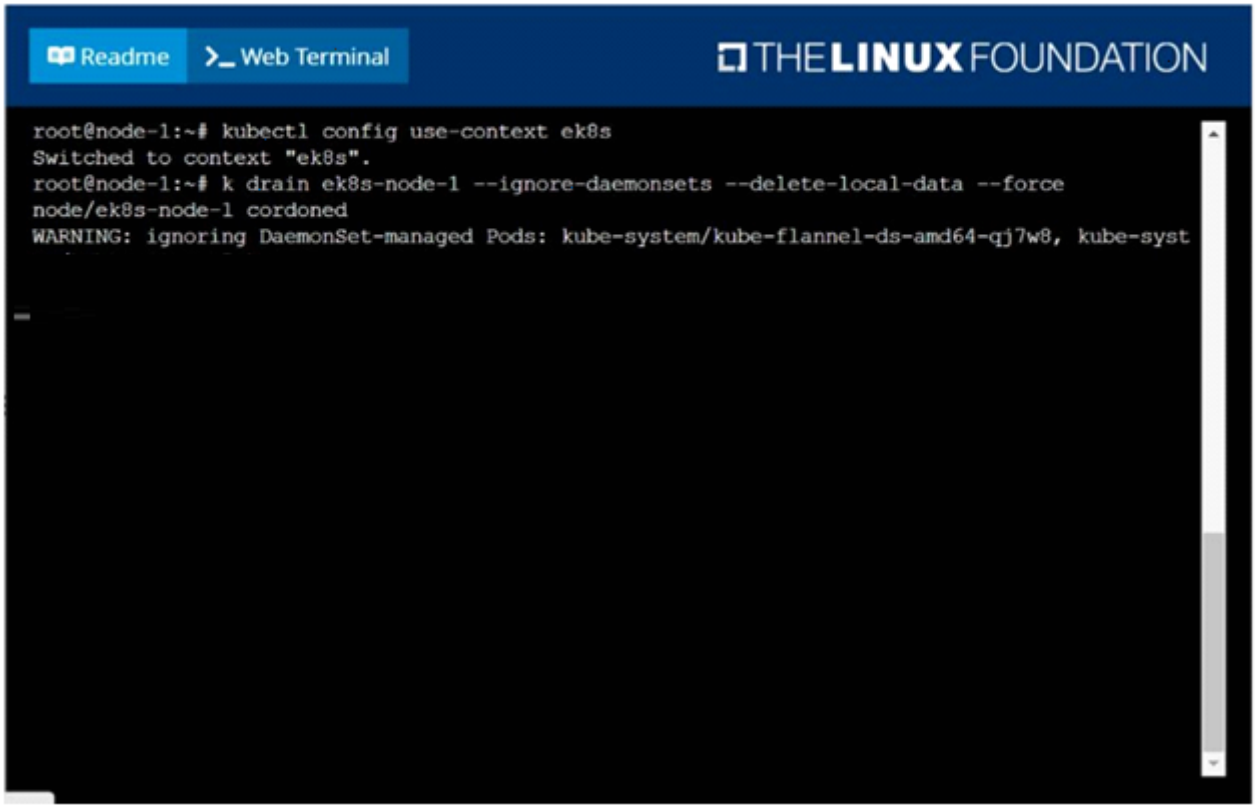
Set the node named ek8s-node-1 as unavailable and reschedule all the pods running on it.

## Options:

**A)** solution


Readme
Web Terminal
THE LINUX FOUNDATION
root@node-1:~# kubectl config use-context ek8s
Switched to context "ek8s".
root@node-1:~# k drain ek8s-node-1 --ignore-daemonsets --delete-local-data --force
node/ek8s-node-1 cordoned
WARNING: ignoring DaemonSet-managed Pods: kube-system/kube-flannel-ds-amd64-qj7w8, kube-syst

**B)** solution




### Answer:

A

## Question 9

**Question Type:** MultipleChoice

SIMULATION

Create a deployment as follows:

* Name: nginx-random

* Exposed via a service nginx-random

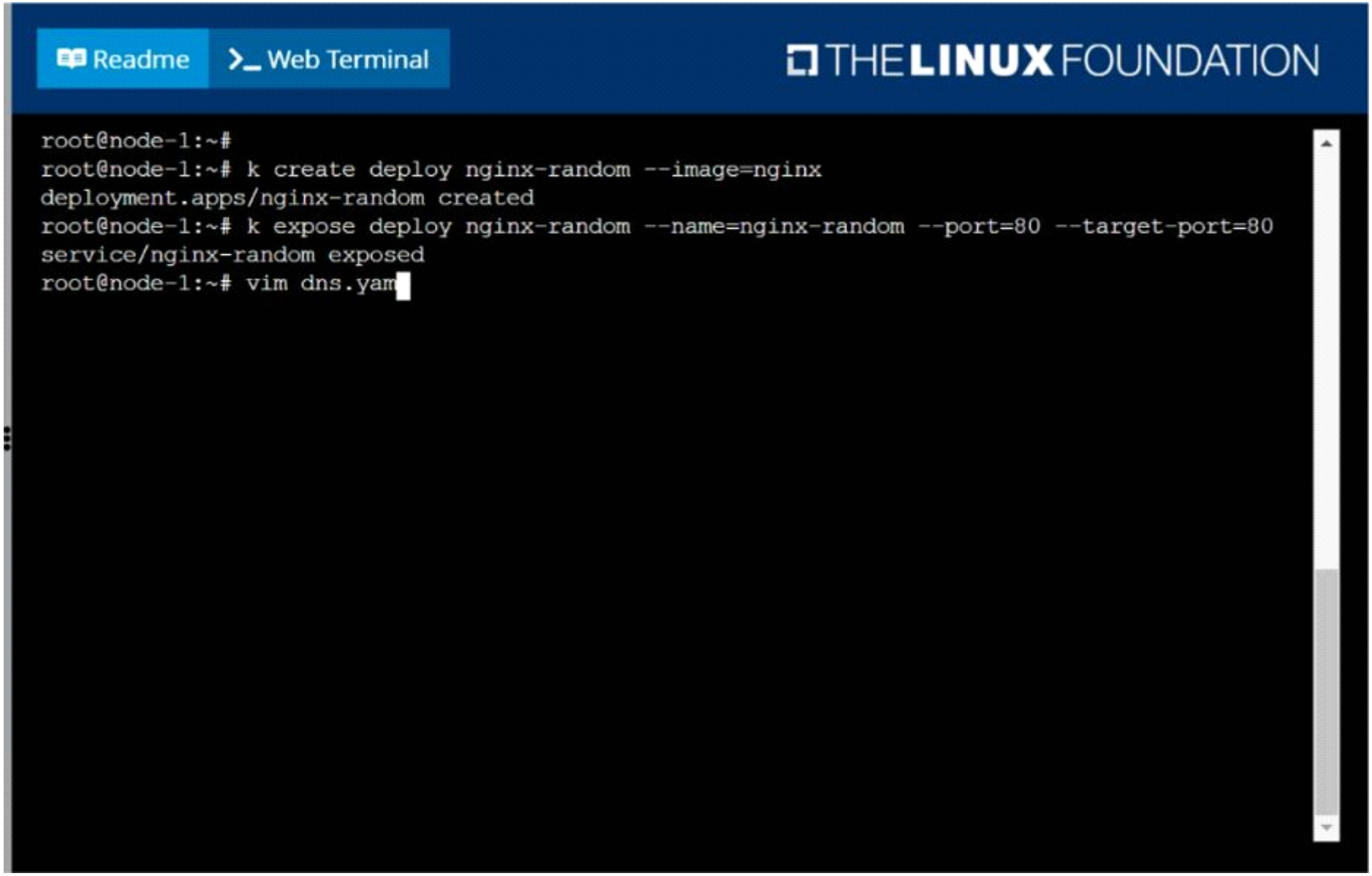
* Ensure that the service & pod are accessible via their respective DNS records

* 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.

## Options:

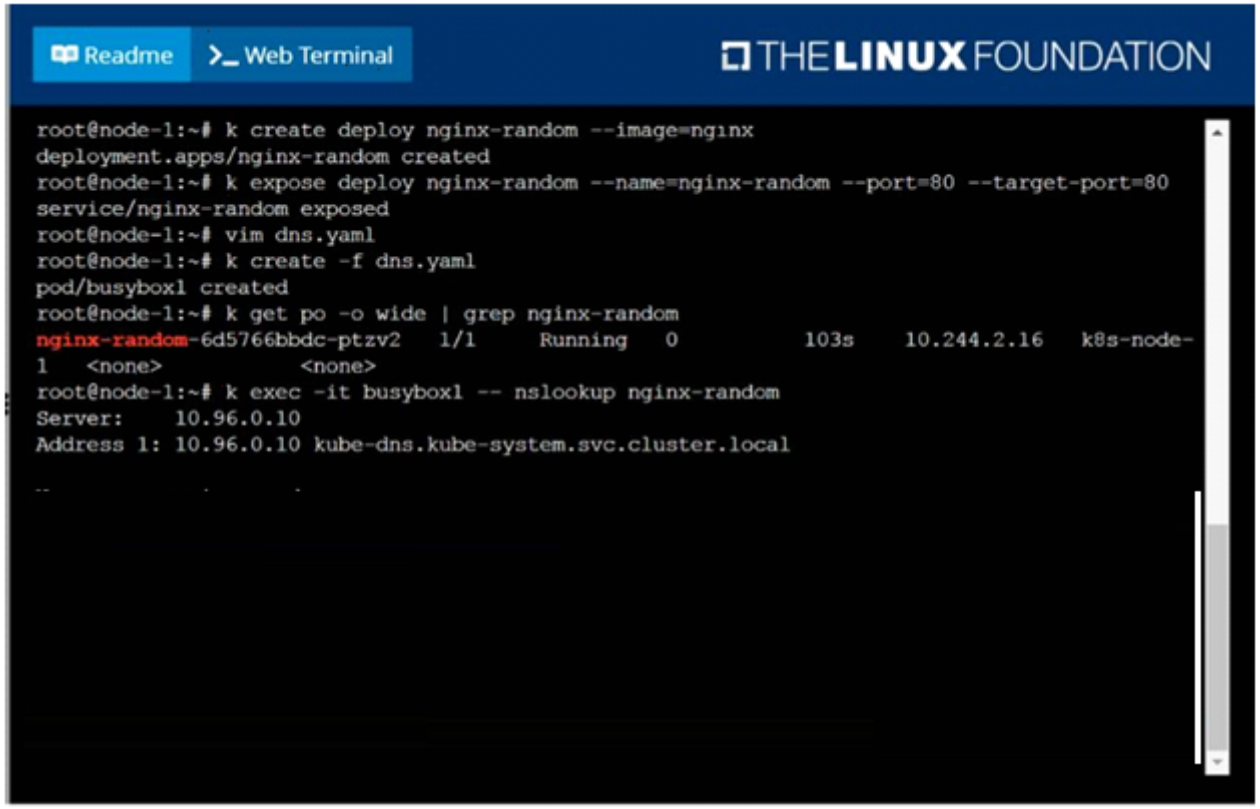
**A)** Solution:


Readme
Web Terminal
THE LINUX FOUNDATION
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.yam

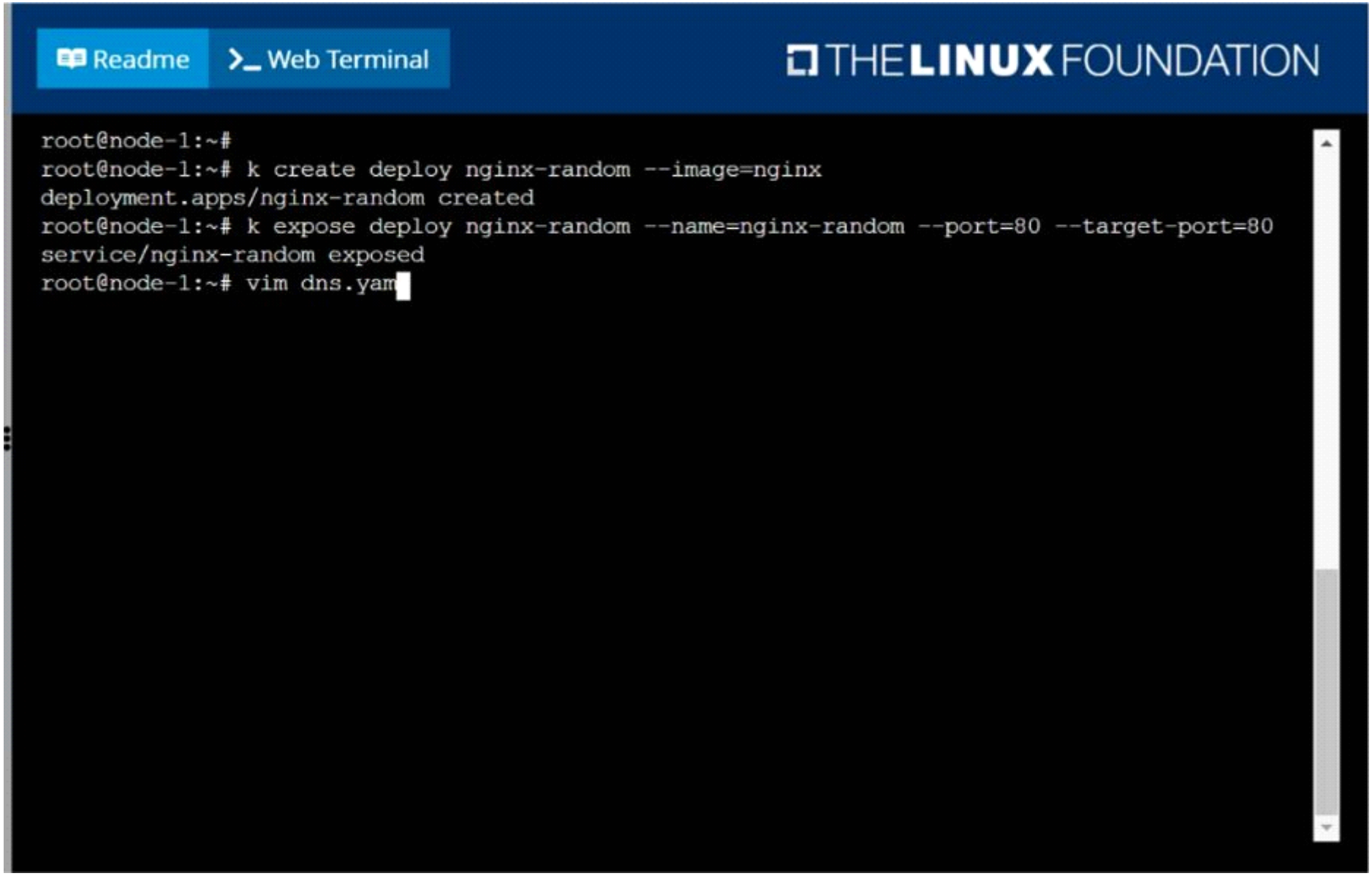
Readme | Web Terminal

THE LINUX FOUNDATION

```
apiVersion: v1
kind: Pod
metadata:
  name: busybox1
  labels:
    name: busybox
spec:
  containers:
  - image: busybox:1.28
    command:
      - sleep
      - "3600"
    name: busybox
~
~
~
~
~
~
~
~
~
~
~
~
:
```


Readme
Web Terminal
THE LINUX FOUNDATION
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
root@node-1:~# k create -f dns.yaml
pod/busybox1 created
root@node-1:~# k get po -o wide | grep nginx-random
nginx-random-6d5766bbdc-ptzv2 1/1 Running 0 103s 10.244.2.16 k8s-node-
1 <none> <none>
root@node-1:~# k exec -it busybox1 -- nslookup nginx-random
Server: 10.96.0.10
Address 1: 10.96.0.10 kube-dns.kube-system.svc.cluster.local

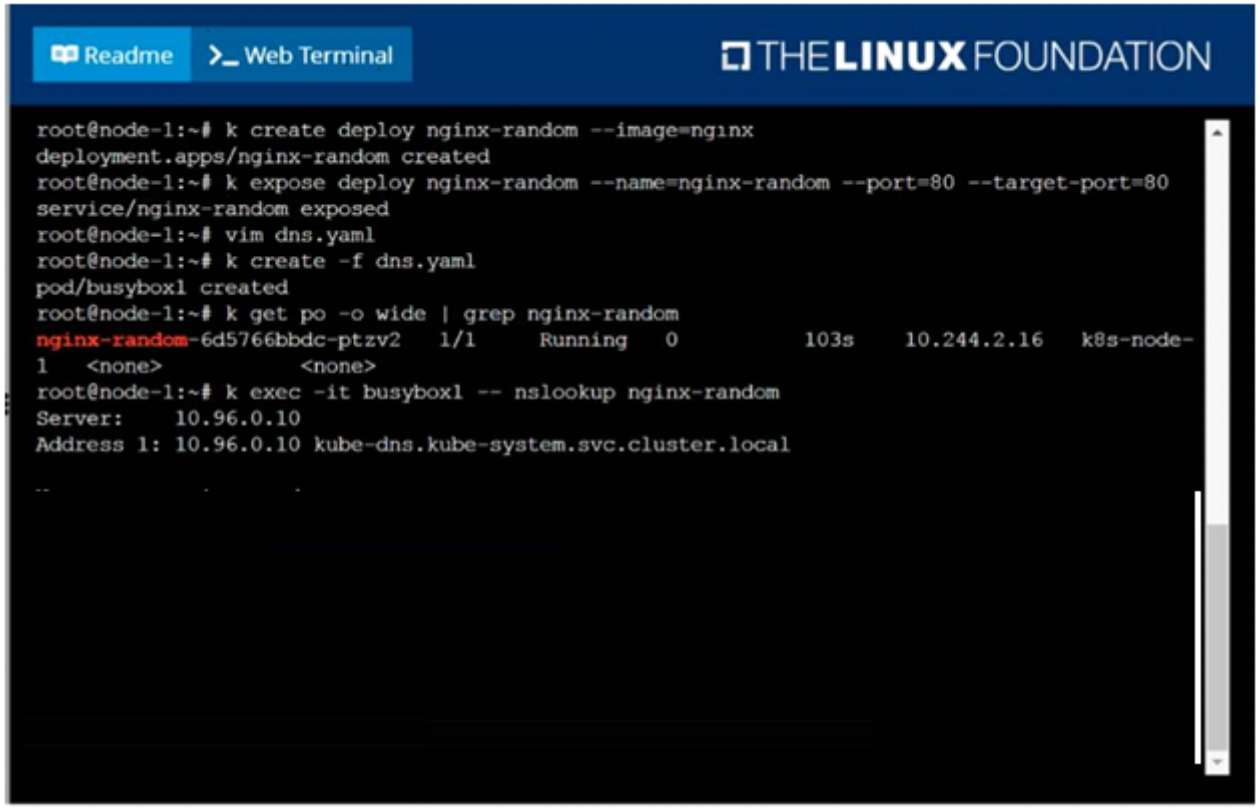
**B)** Solution:


Readme
Web Terminal
THE LINUX FOUNDATION
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.yam

Readme | Web Terminal

THE LINUX FOUNDATION

```
apiVersion: v1
kind: Pod
metadata:
  name: busybox1
  labels:
    name: busybox
spec:
  containers:
  - image: busybox:1.28
    command:
      - sleep
      - "3600"
    name: busybox
~
~
~
~
~
~
~
~
~
~
~
~
:
```




### Answer:

A

## Question 10

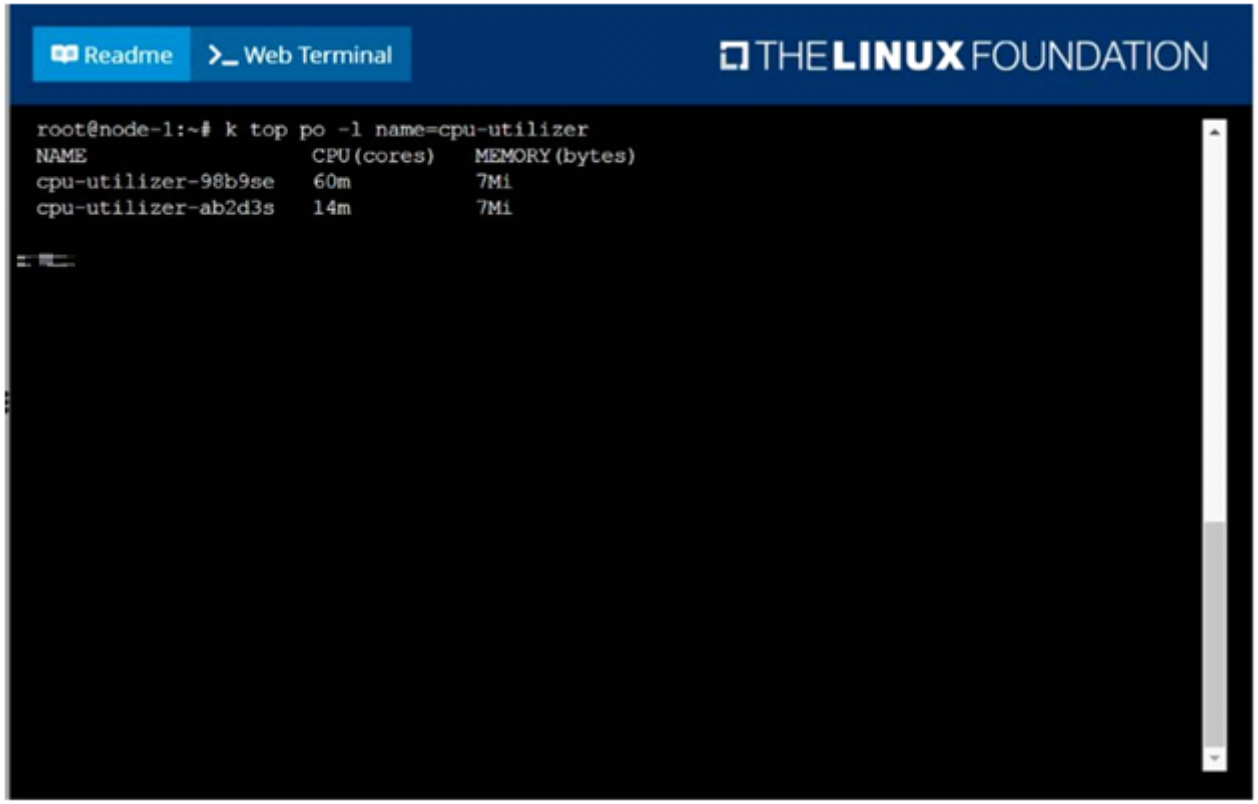
**Question Type:** MultipleChoice

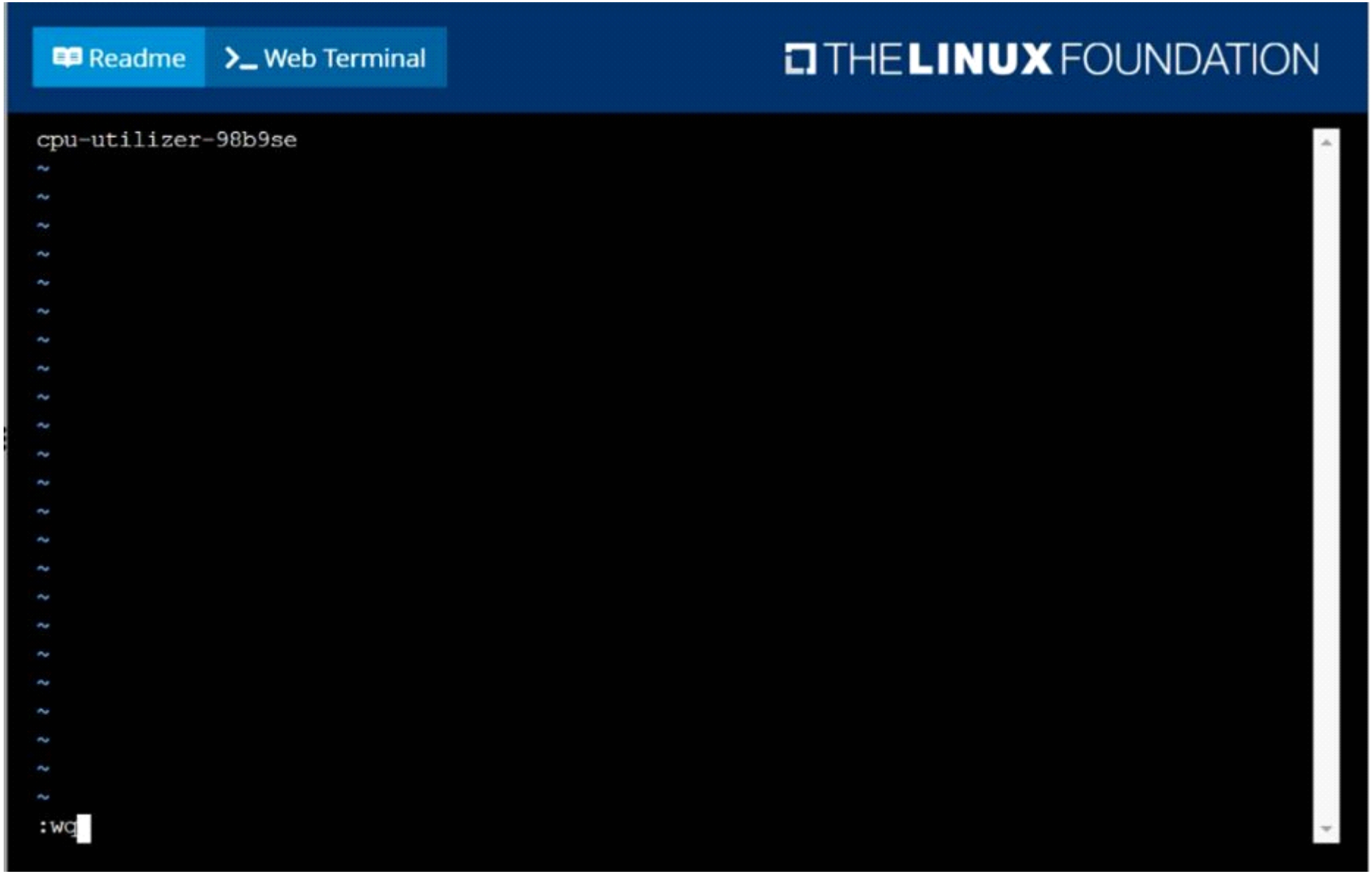
SIMULATION

From the pod label name=cpu-utilizer, find pods running high CPU workloads and write the name of the pod consuming most CPU to the file /opt/KUTR00102/KUTR00102.txt (which already exists).

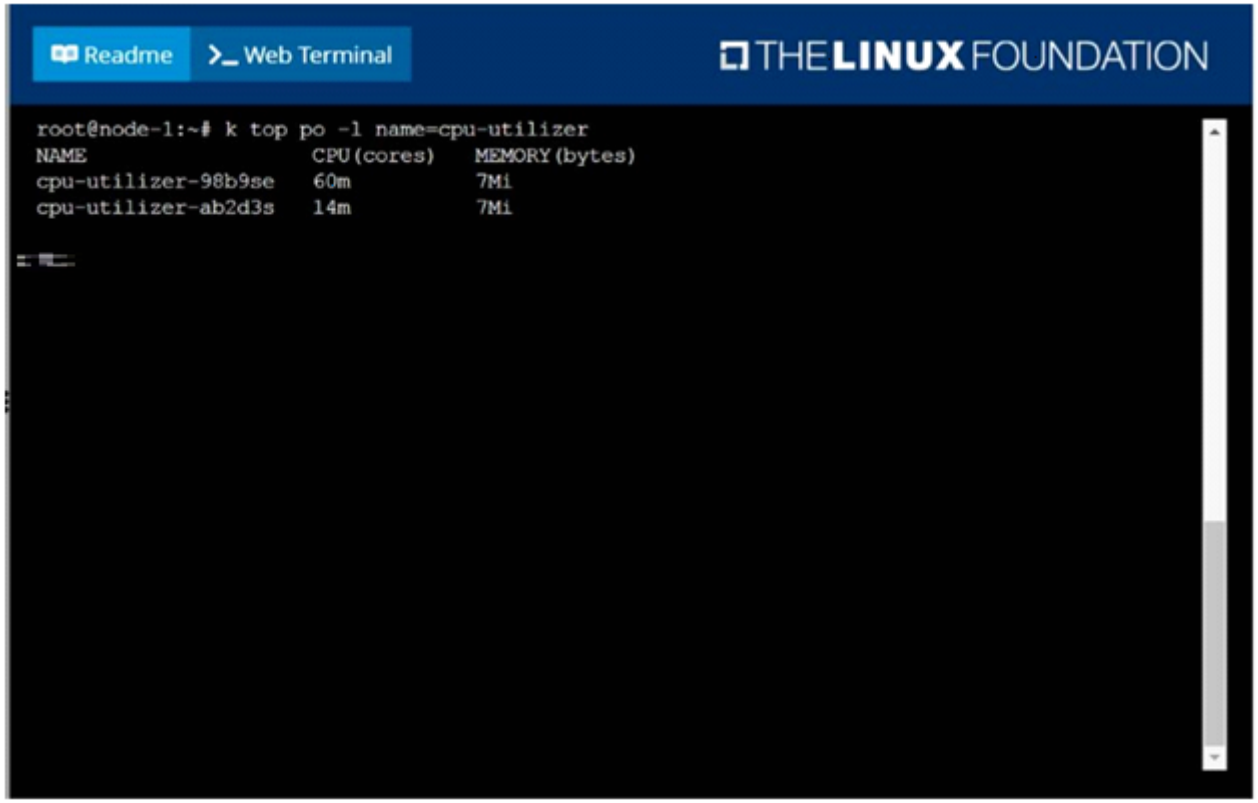
## Options:

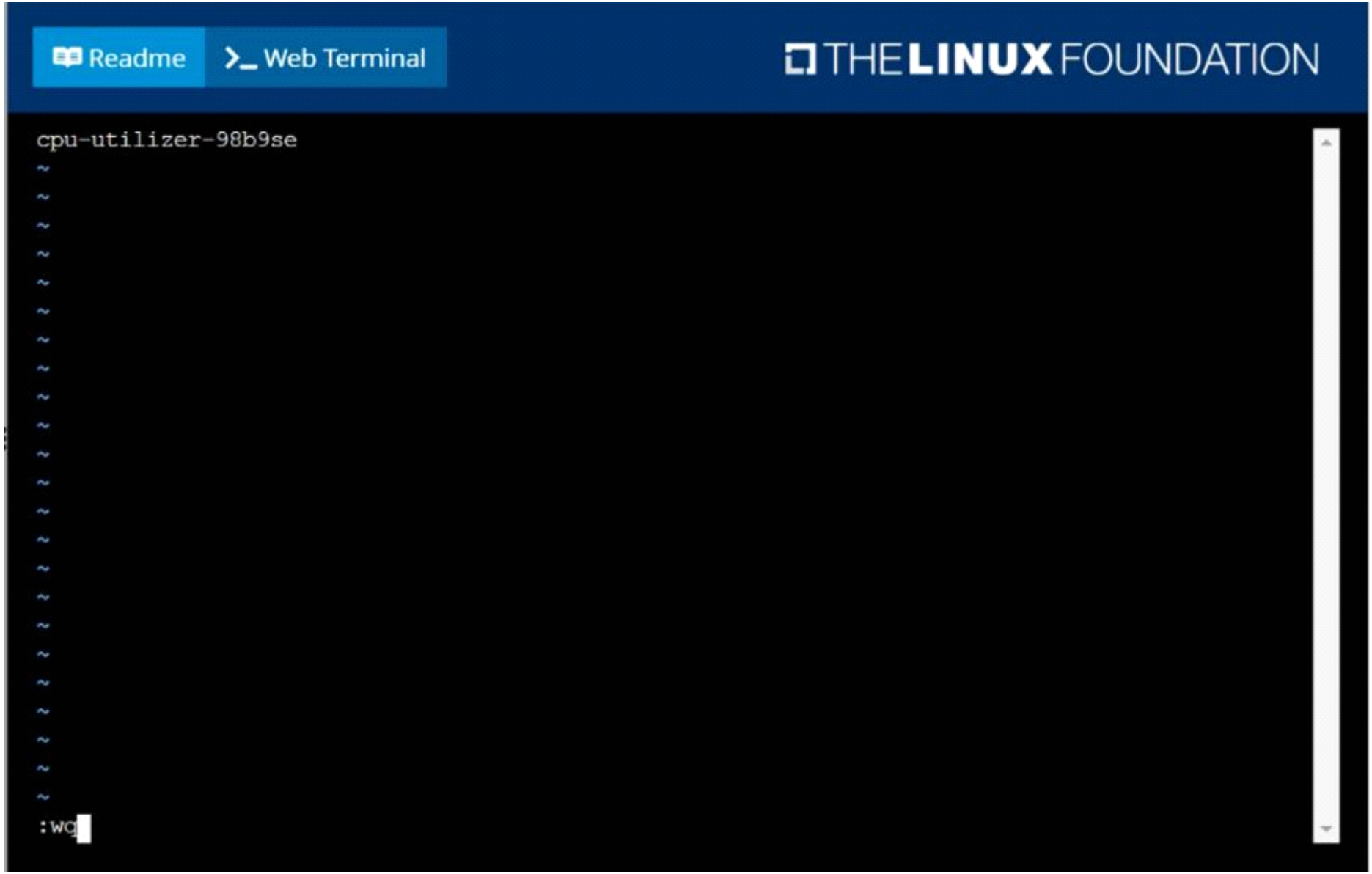
**A)** solution


Readme
Web Terminal
THE LINUX FOUNDATION
root@node-1:~# k top po -l name=cpu-utilizer
NAME CPU(cores) MEMORY(bytes)
cpu-utilizer-98b9se 60m 7Mi
cpu-utilizer-ab2d3s 14m 7Mi


Readme
Web Terminal
THE LINUX FOUNDATION
cpu-utilizer-98b9se
:wq

**B)** solution


Readme
Web Terminal
THE LINUX FOUNDATION
root@node-1:~# k top po -l name=cpu-utilizer
NAME CPU(cores) MEMORY(bytes)
cpu-utilizer-98b9se 60m 7Mi
cpu-utilizer-ab2d3s 14m 7Mi


Readme
Web Terminal
THE LINUX FOUNDATION
cpu-utilizer-98b9se
:wq

## Answer:

A