

CKA Dumps

Certified Kubernetes Administrator (CKA) Program

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



NEW QUESTION 1

CORRECT TEXT

List all the pods sorted by name

- A. Mastered
- B. Not Mastered

Answer: A

Explanation:

kubect1 get pods --sort-by=.metadata.name

NEW QUESTION 2

CORRECT TEXT

List pod logs named “frontend” and search for the pattern “started” and write it to a file “/opt/error-logs”

- A. Mastered
- B. Not Mastered

Answer: A

Explanation:

Kubectl logs frontend | grep -i “started” > /opt/error-logs

NEW QUESTION 3

CORRECT TEXT

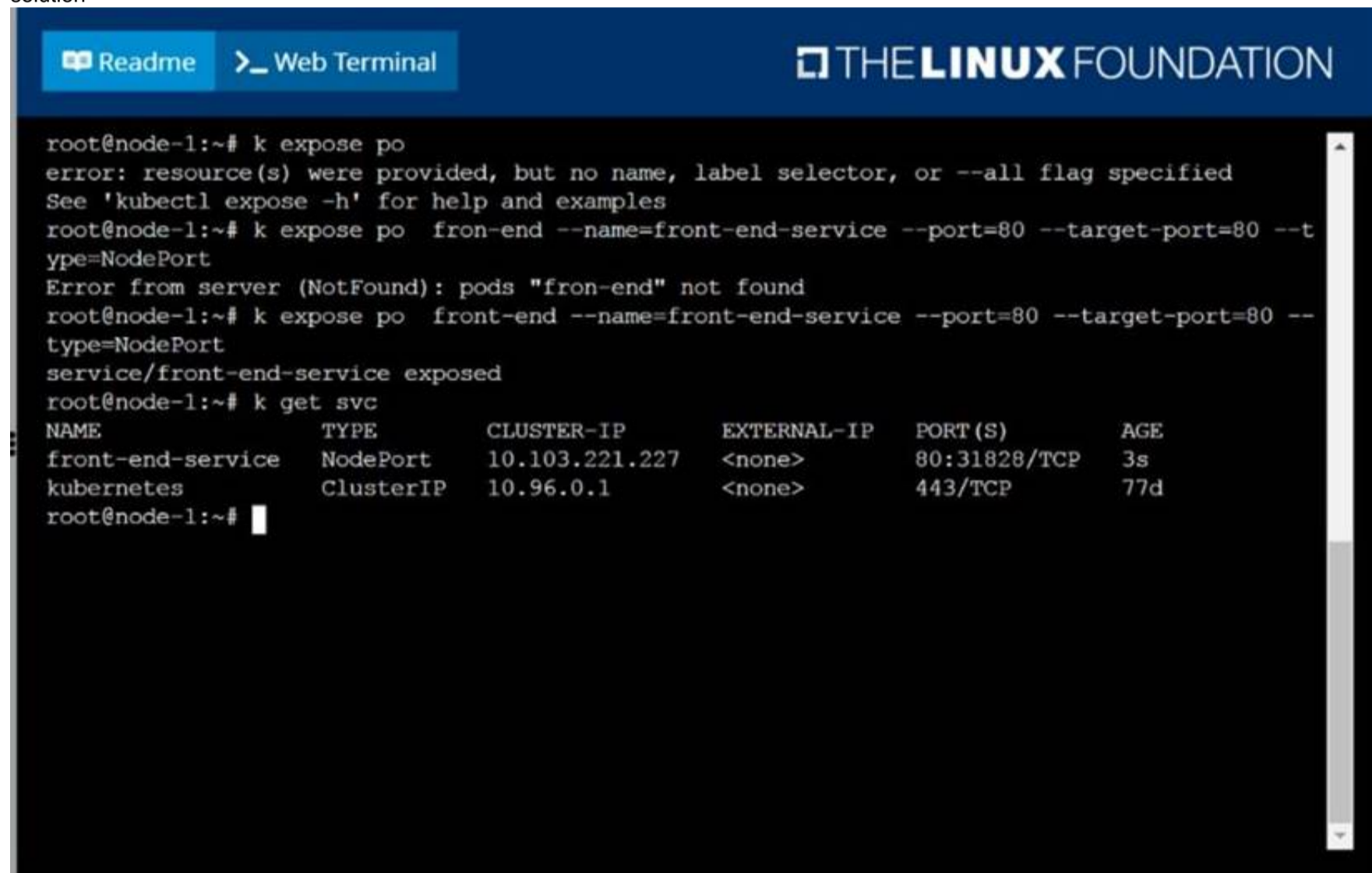
Create and configure the service front-end-service so it's accessible through NodePort and routes to the existing pod named front-end.

- 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 4

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

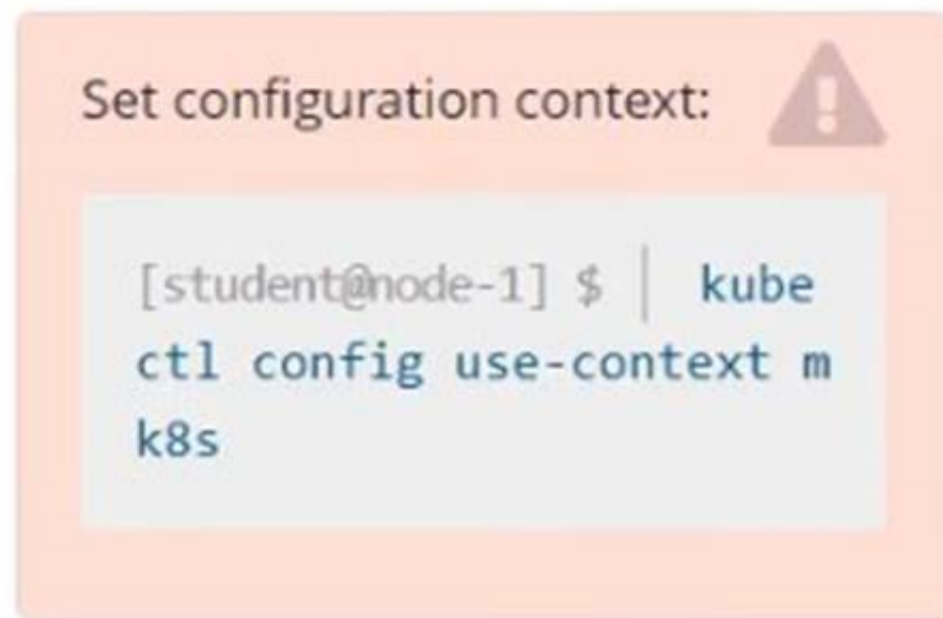
kubect1 get pods -o=jsonpath='{range

items[*]}{.metadata.name}{"\t"}{.status.podIP}{"\n"}{end}'

NEW QUESTION 5

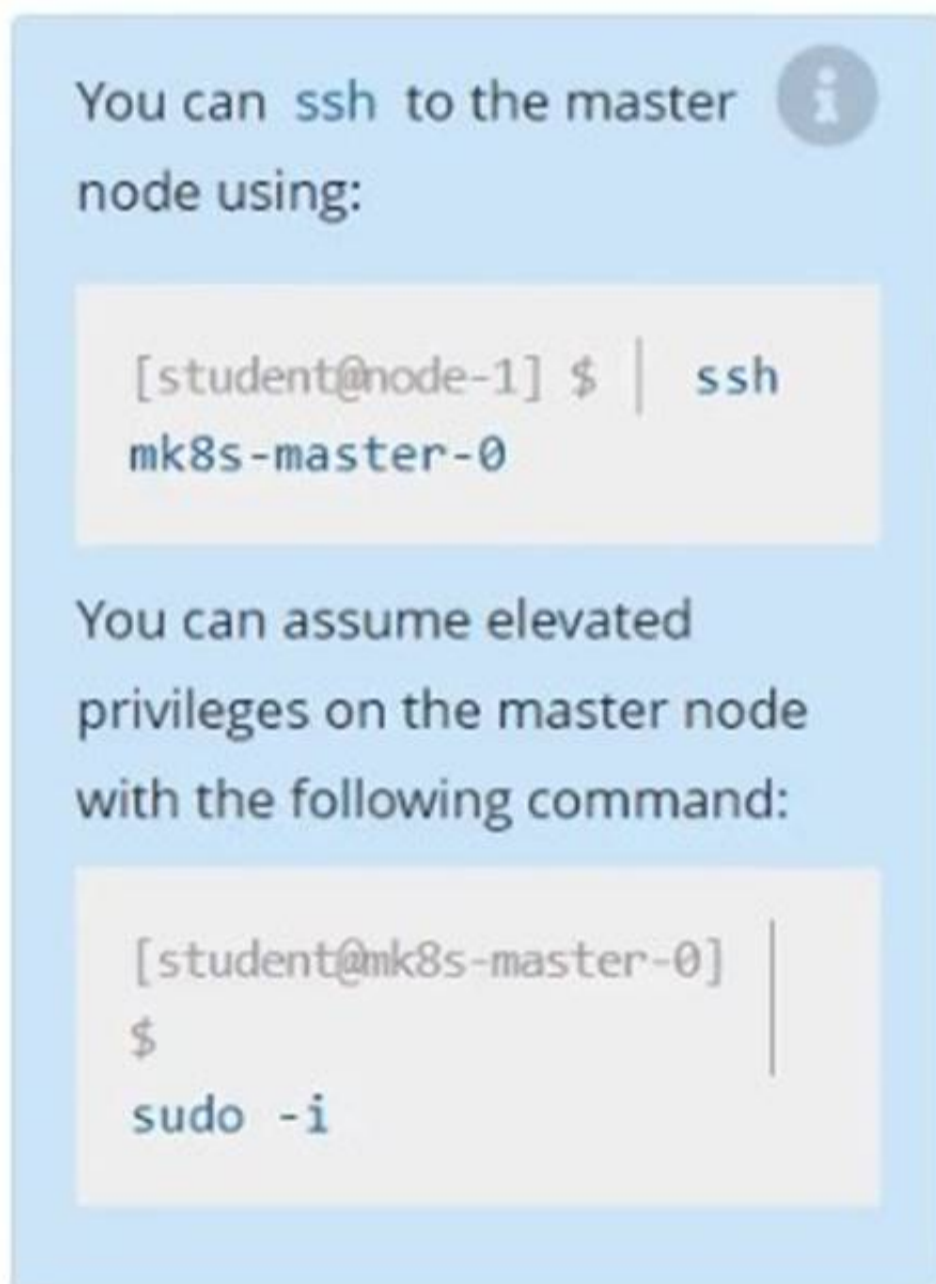
CORRECT TEXT

Score: 7%

**Task**

Given an existing Kubernetes cluster running version 1.20.0, upgrade all of the Kubernetes control plane and node components on the master node only to version 1.20.1.

Be sure to drain the master node before upgrading it and uncordon it after the upgrade.



You are also expected to upgrade kubelet and kubect1 on the master node.

Do not upgrade the worker nodes, etcd, the container manager, the CNI plugin, the DNS service or any other addons.

- A. Mastered
B. Not Mastered

Answer: A

Explanation:

SOLUTION:

```
[student@node-1] > ssh ek8s
kubectl cordon k8s-master
kubectl drain k8s-master --delete-local-data --ignore-daemonsets --force
apt-get install kubeadm=1.20.1-00 kubelet=1.20.1-00 kubectl=1.20.1-00 --
disableexcludes=kubernetes
kubeadm upgrade apply 1.20.1 --etcd-upgrade=false
systemctl daemon-reload
systemctl restart kubelet kubectl
uncordon k8s-master
```

NEW QUESTION 6

CORRECT TEXT

Score: 5%

Set configuration context:

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

Task

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/KUTR00401/KUTR00401.txt (which already exists).

- A. Mastered
B. Not Mastered

Answer: A

Explanation:

Solution:

```
kubectl top -l name=cpu-user -A
echo 'pod name' >> /opt/KUT00401/KUT00401.txt
```

NEW QUESTION 7

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

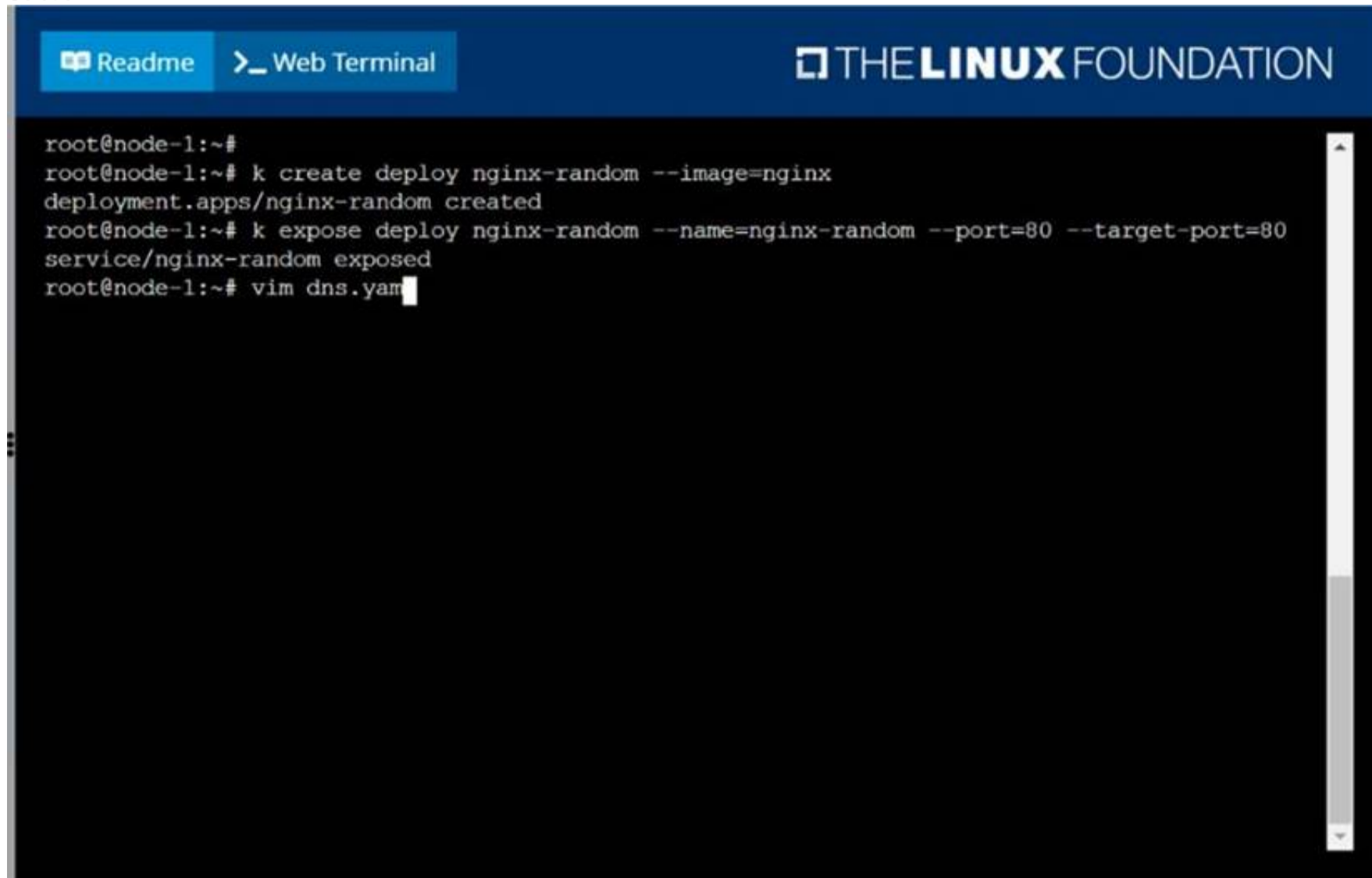
Next, use the utility nslookup to look up the DNS records of the service & pod and write the output to /opt/KUNW00601/service.dns and /opt/KUNW00601/pod.dns respectively.

- A. Mastered
- B. Not Mastered

Answer: A

Explanation:

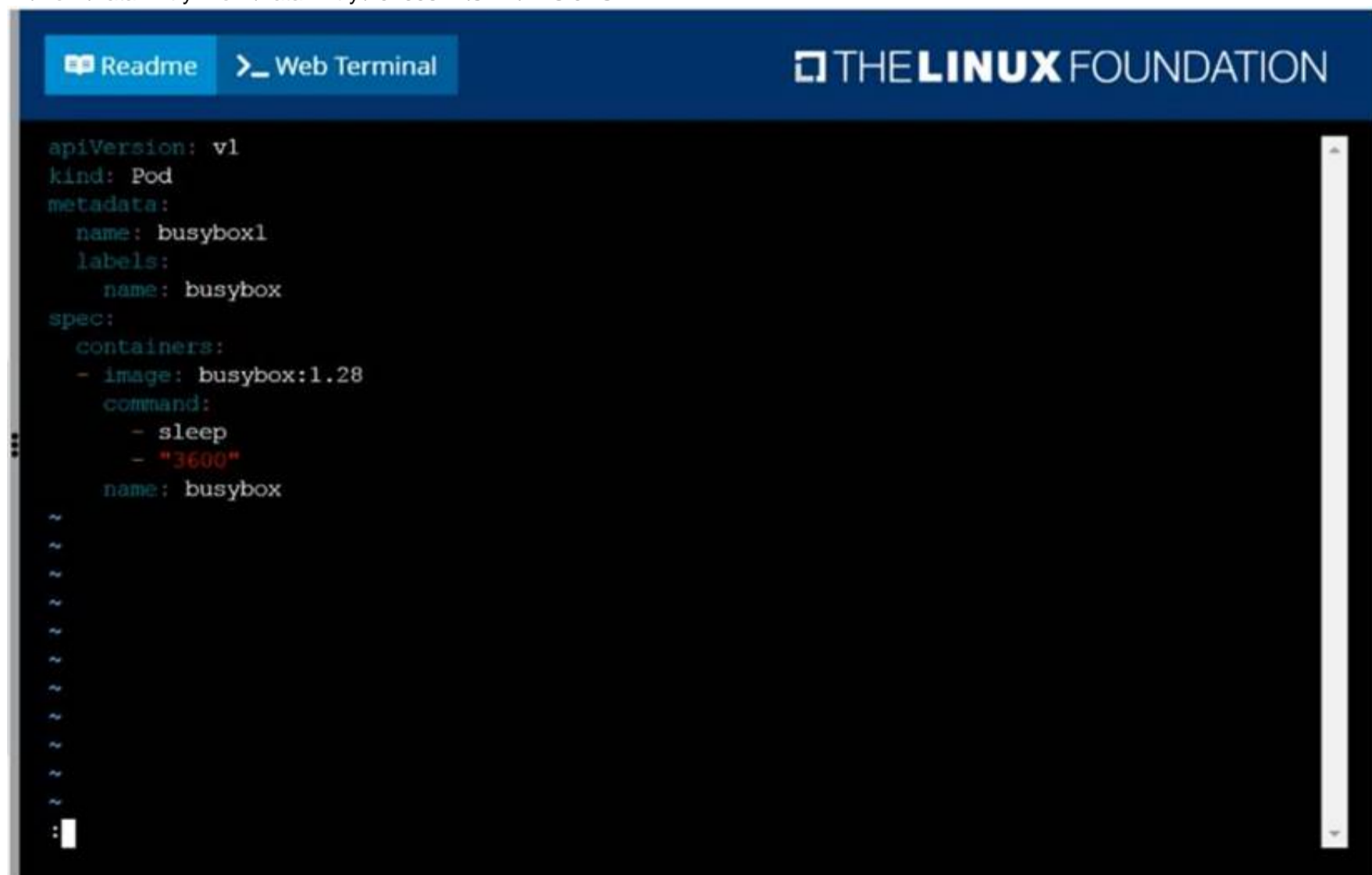
Solution:



The screenshot shows a web terminal interface with a dark background. At the top, there is a blue header bar with the text "THE LINUX FOUNDATION" on the right. Below the header, there are two tabs: "Readme" and "Web Terminal". The "Web Terminal" tab is active, displaying a terminal session. The terminal output shows the following commands and their results:

```
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
```

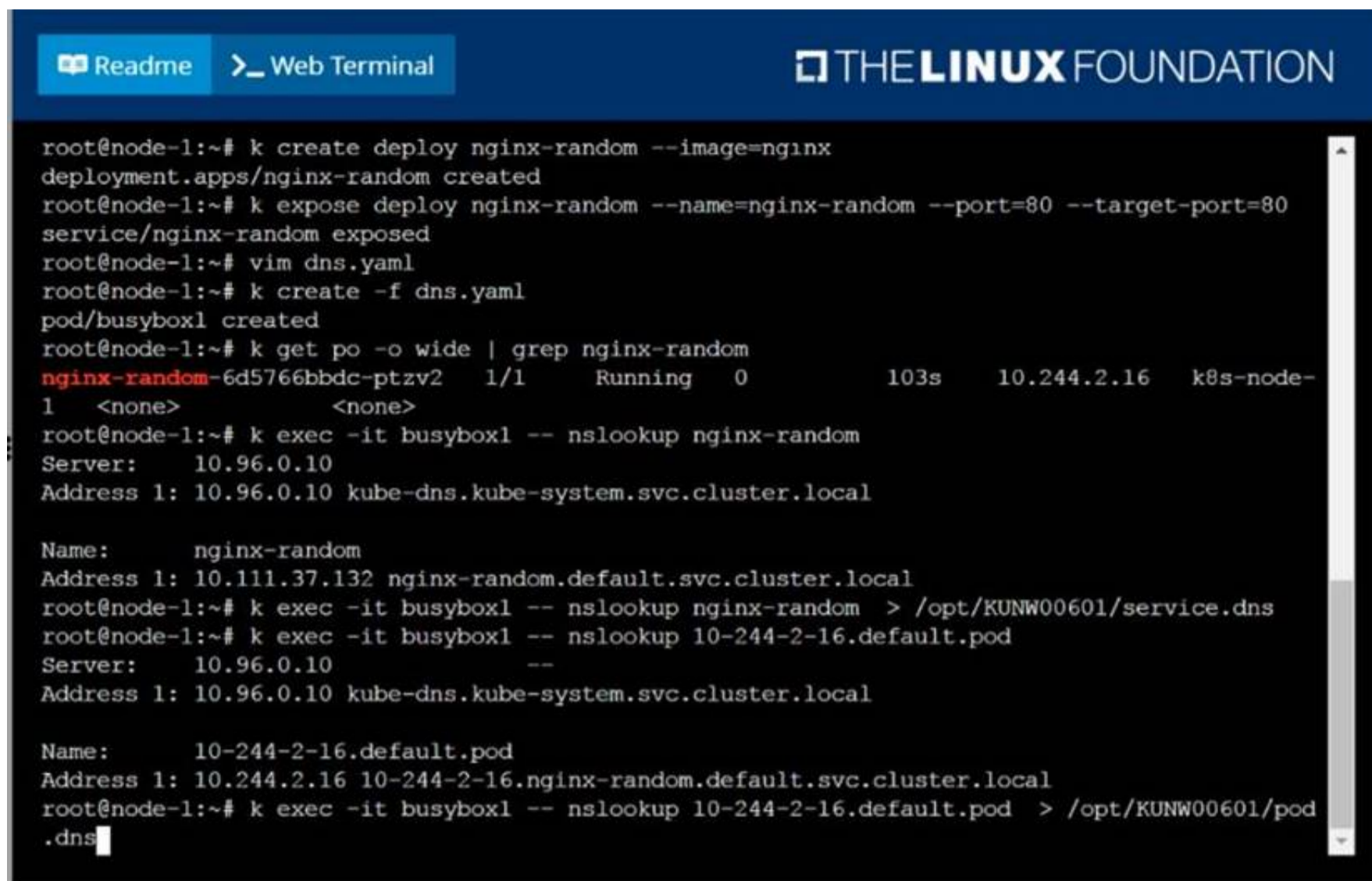
F:\Work\Data Entry Work\Data Entry\20200827\CKA\17 C.JPG



The screenshot shows a web terminal interface with a dark background. At the top, there is a blue header bar with the text "THE LINUX FOUNDATION" on the right. Below the header, there are two tabs: "Readme" and "Web Terminal". The "Web Terminal" tab is active, displaying a terminal session. The terminal output shows the following YAML manifest:

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

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



```
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

Name:      nginx-random
Address 1: 10.111.37.132 nginx-random.default.svc.cluster.local
root@node-1:~# k exec -it busybox1 -- nslookup nginx-random > /opt/KUNW00601/service.dns
root@node-1:~# k exec -it busybox1 -- nslookup 10-244-2-16.default.pod
Server:      10.96.0.10
Address 1: 10.96.0.10 kube-dns.kube-system.svc.cluster.local

Name:      10-244-2-16.default.pod
Address 1: 10.244.2.16 10-244-2-16.nginx-random.default.svc.cluster.local
root@node-1:~# k exec -it busybox1 -- nslookup 10-244-2-16.default.pod > /opt/KUNW00601/pod.dns
```

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

NEW QUESTION 8

CORRECT TEXT

Create a deployment spec file that will:

? Launch 7 replicas of the nginx Image with the labelapp_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).

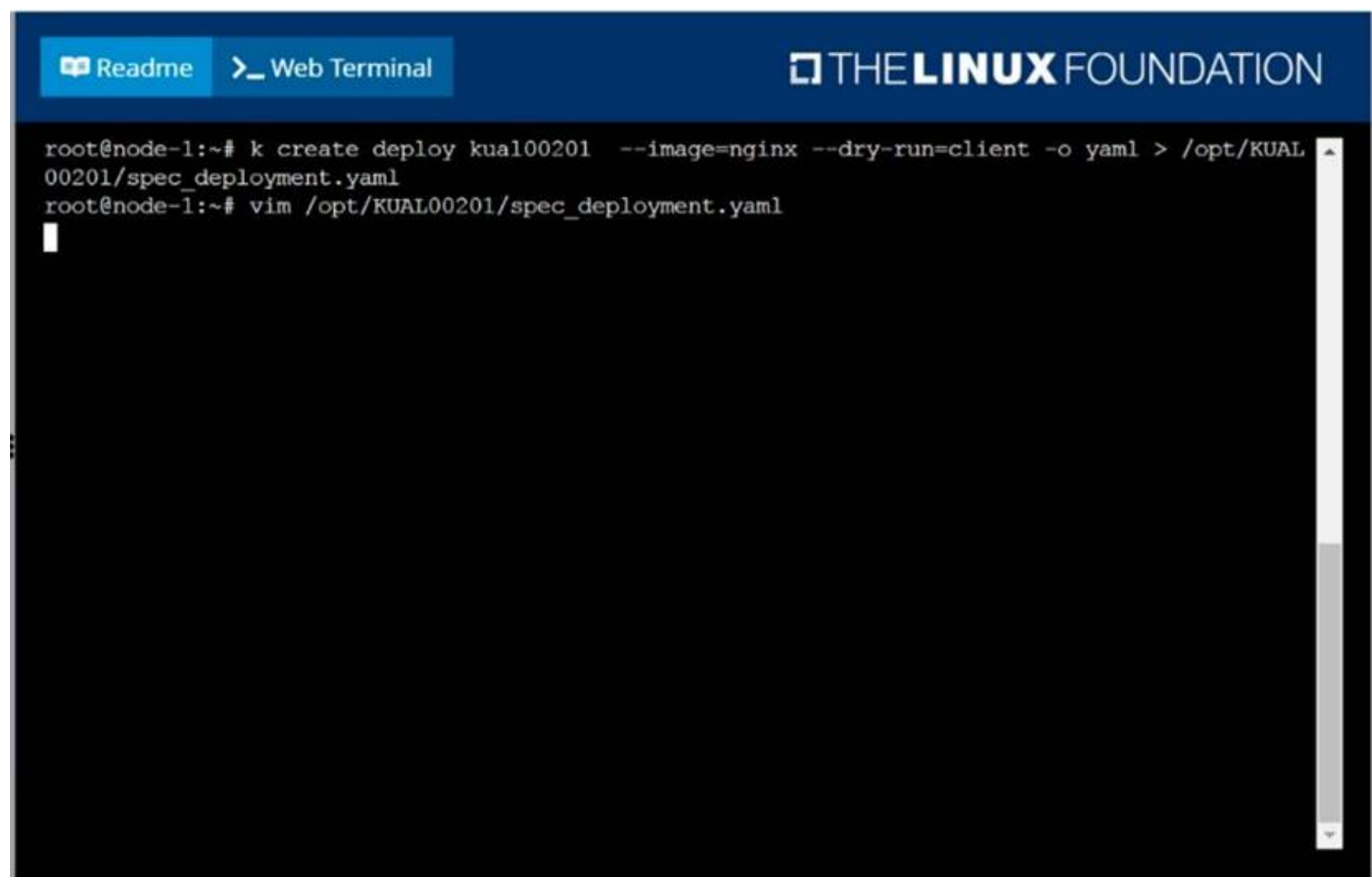
When you are done, clean up (delete) any new Kubernetes API object that you produced during this task.

- A. Mastered
- B. Not Mastered

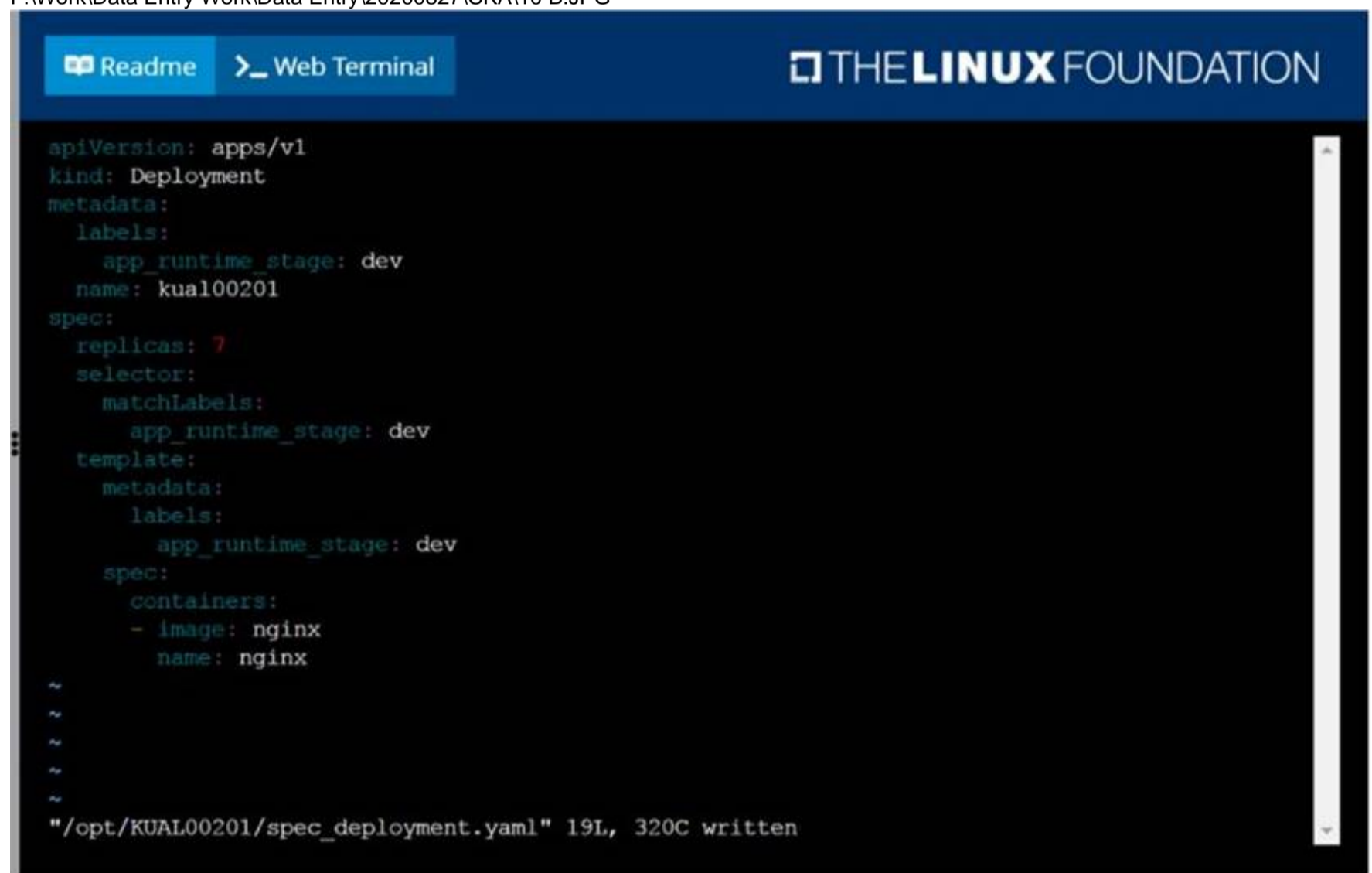
Answer: A

Explanation:

solution



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




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

NEW QUESTION 9

CORRECT TEXT


Score: 7%

No configuration context change required for this task. 

Ensure, however, that you have returned to the base node before starting to work on this task:

```
[student@mk8s-master-0] |
$
exit
```

Task
First, create a snapshot of the existing etcd instance running at <https://127.0.0.1:2379>, saving the snapshot to `/srv/data/etcd-snapshot.db`.

Creating a snapshot of the given instance is expected to complete in seconds. 

If the operation seems to hang, something's likely wrong with your command. Use **CTRL + C** to cancel the operation and try again.

Next, restore an existing, previous snapshot located at `/var/lib/backup/etcd-snapshot-previous.db`

The following TLS certificates/key are supplied for connecting to the server with etcdctl :

- CA certificate:
/opt/KUIN00601/ca.crt
- Client certificate:
/opt/KUIN00601/etcd-client.crt
- Client key:
/opt/KUIN00601/etcd-client.key

- A. Mastered
B. Not Mastered

Answer: A

Explanation:

Solution:

#backup

ETCDCTL_API=3 etcdctl --endpoints="https://127.0.0.1:2379" --

cacert=/opt/KUIN000601/ca.crt --cert=/opt/KUIN000601/etcd-client.crt -- key=/opt/KUIN000601/etcd-client.key snapshot save /etc/data/etcd-snapshot.db

#restore


ETCDCTL_API=3 etcdctl --endpoints="https://127.0.0.1:2379" --

cacert=/opt/KUIN000601/ca.crt --cert=/opt/KUIN000601/etcd-client.crt -- key=/opt/KUIN000601/etcd-client.key snapshot restore /var/lib/backup/etcd-snapshot-previoys.db

NEW QUESTION 10

CORRECT TEXT

Score: 7%


Set configuration context: 

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

Task

Create a new nginx Ingress resource as follows:

- Name: ping
- Namespace: ing-internal
- Exposing service hi on path /hi using service port 5678

The availability of service hi 
can be checked using the
following command, which
should return hi :

```
[student@node-1] $ | curl  
-kL <INTERNAL_IP>/hi
```

- A. Mastered
B. Not Mastered

Answer: A

Explanation:

Solution:
vi ingress.yaml

apiVersion: networking.k8s.io/v1
kind: Ingress
metadata:
name: ping
namespace: ing-internal
spec:
rules:
- http:
paths:
- path: /hi
pathType: Prefix
backend:
service:
name: hi
port:
number: 5678

kubectl create -f ingress.yaml

NEW QUESTION 10

CORRECT TEXT

Create a snapshot of the etcd instance running at <https://127.0.0.1:2379>, saving the snapshot to the file path /srv/data/etcd-snapshot.db.

The following TLS certificates/key are supplied for connecting to the server with etcdctl:

- ? CA certificate: /opt/KUCM00302/ca.crt
- ? Client certificate: /opt/KUCM00302/etcd-client.crt
- ? Client key: /opt/KUCM00302/etcd-client.key

- A. Mastered
B. Not Mastered

Answer: A

Explanation:

solution

Readme
Web Terminal

```

root@node-1:~# ETCDCCTL_API=3 etcdctl --endpoints=https://127.0.0.1:2379 --cacert=/opt/KUCM00302/ca.crt --cert=/opt/KUCM00302/etcd-client.crt --key=/opt/KUCM00302/etcd-client.key snapshot save /srv/data/etcd-snapshot.db
{"level":"info","ts":1598530470.8313155,"caller":"snapshot/v3_snapshot.go:110","msg":"create d temporary db file","path":"/srv/data/etcd-snapshot.db.part"}
{"level":"warn","ts":"2020-08-27T12:14:30.838Z","caller":"clientv3/retry_interceptor.go:116","msg":"retry stream intercept"}
{"level":"info","ts":1598530470.8388612,"caller":"snapshot/v3_snapshot.go:121","msg":"fetching snapshot","endpoint":"https://127.0.0.1:2379"}
{"level":"info","ts":1598530470.8570414,"caller":"snapshot/v3_snapshot.go:134","msg":"fetched snapshot","endpoint":"https://127.0.0.1:2379","took":0.025676157}
{"level":"info","ts":1598530470.8571067,"caller":"snapshot/v3_snapshot.go:143","msg":"saved","path":"/srv/data/etcd-snapshot.db"}
Snapshot saved at /srv/data/etcd-snapshot.db
root@node-1:~#

```

F:\Work\Data Entry Work\Data Entry\20200827\CKA\18 C.JPG

NEW QUESTION 15

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 18**CORRECT TEXT**

For this item, you will have to ssh to the nodes ik8s-master-0 and ik8s-node-0 and complete all tasks on these nodes. Ensure that you return to the base node (hostname: node-1) when you have completed this item.

Context

As an administrator of a small development team, you have been asked to set up a Kubernetes cluster to test the viability of a new application.

Task

You must use kubeadm to perform this task. Any kubeadm invocations will require the use of the --ignore-preflight-errors=all option.

? Configure the node ik8s-master-0 as a master node. .

? Join the node ik8s-node-0 to the cluster.

- A. Mastered
- B. Not Mastered

Answer: A

Explanation:**solution**

You must use the kubeadm configuration file located at /etc/kubeadm.conf when initializing your cluster.

You may use any CNI plugin to complete this task, but if you don't have your favourite CNI plugin's manifest URL at hand, Calico is one popular option:

<https://docs.projectcalico.org/v3.14/manifests/calico.yaml>

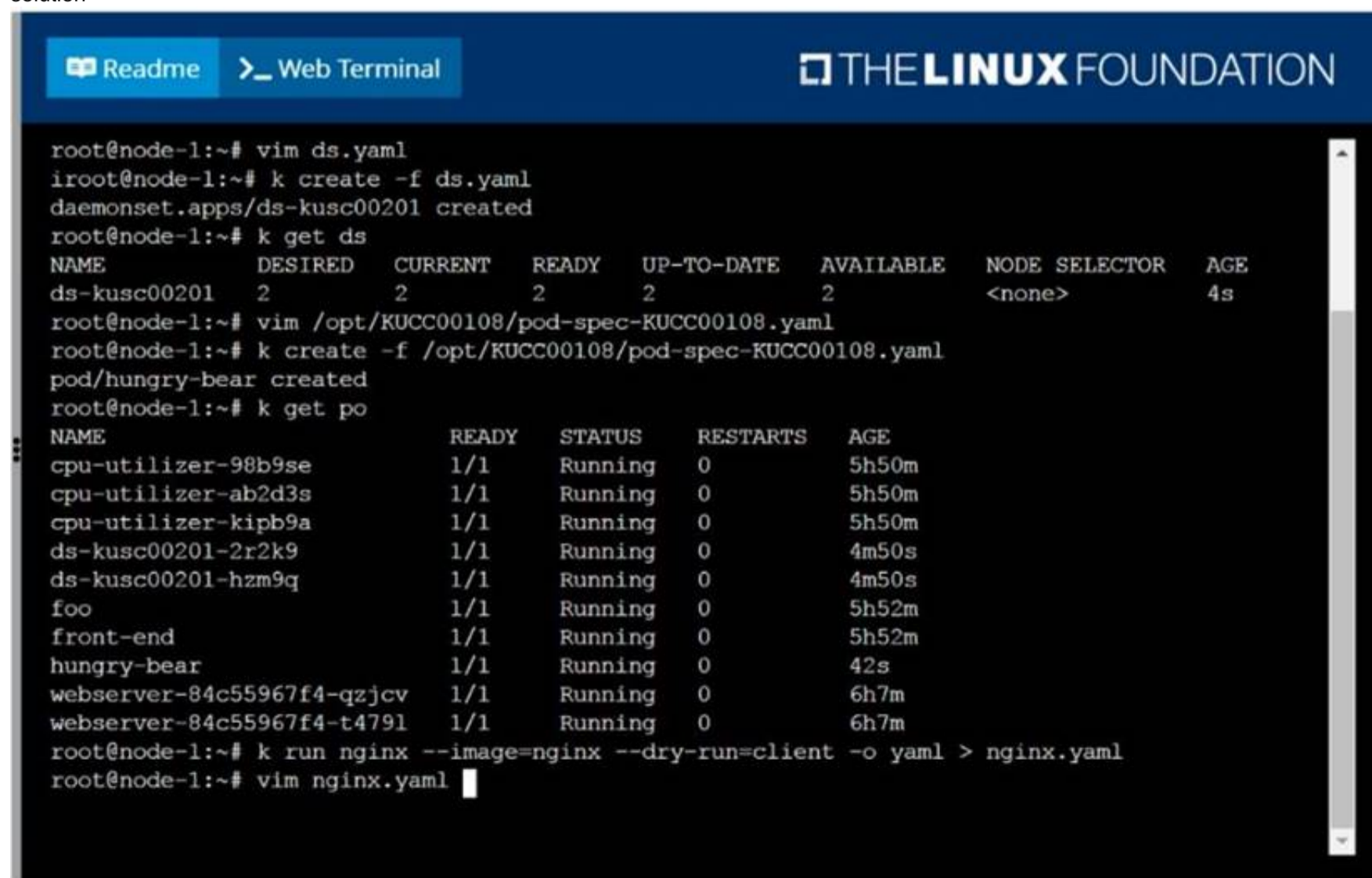
Docker is already installed on both nodes and apt has been configured so that you can install the required tools.

NEW QUESTION 20**CORRECT TEXT**

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.

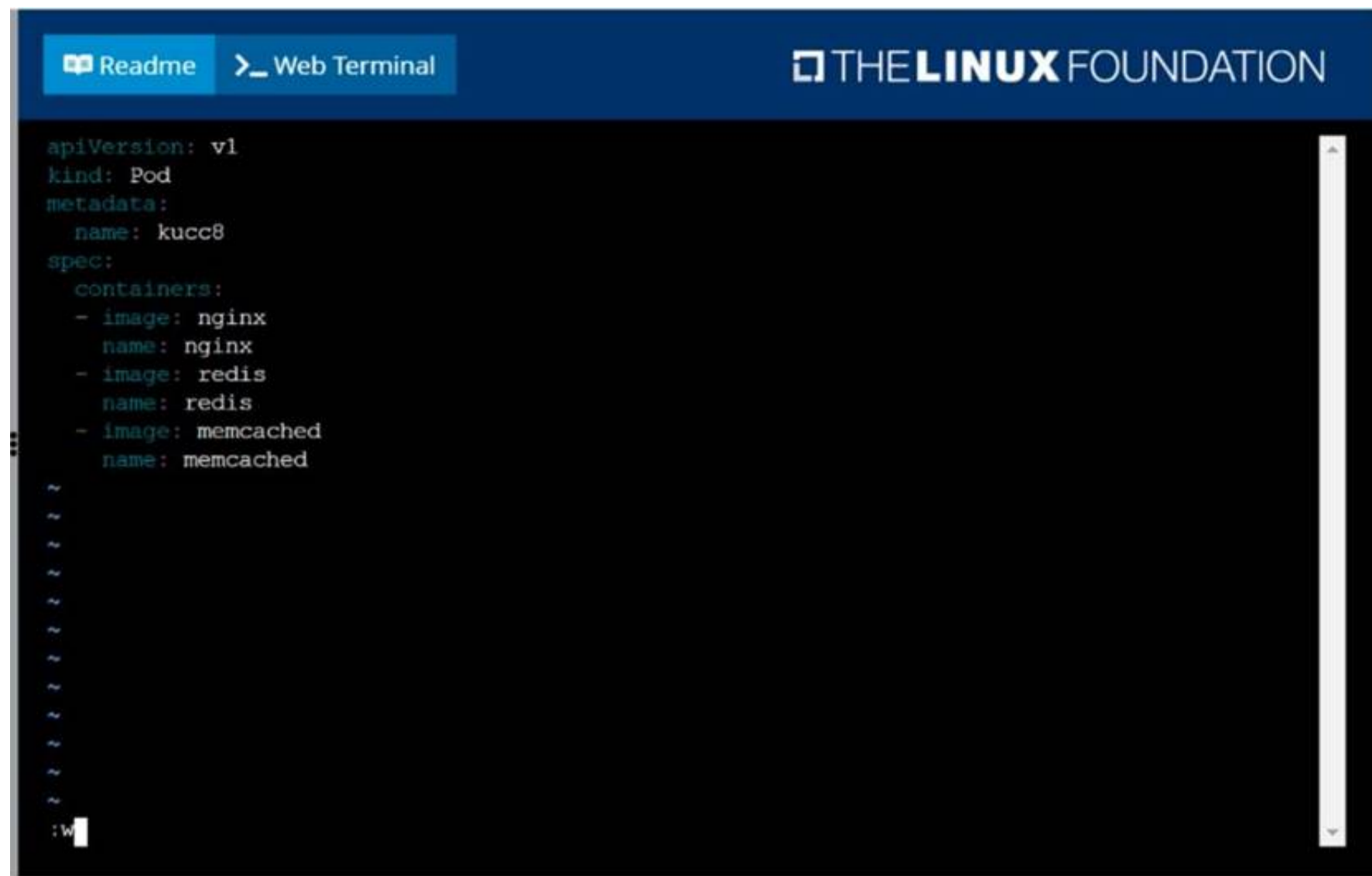
- A. Mastered
- B. Not Mastered

Answer: A

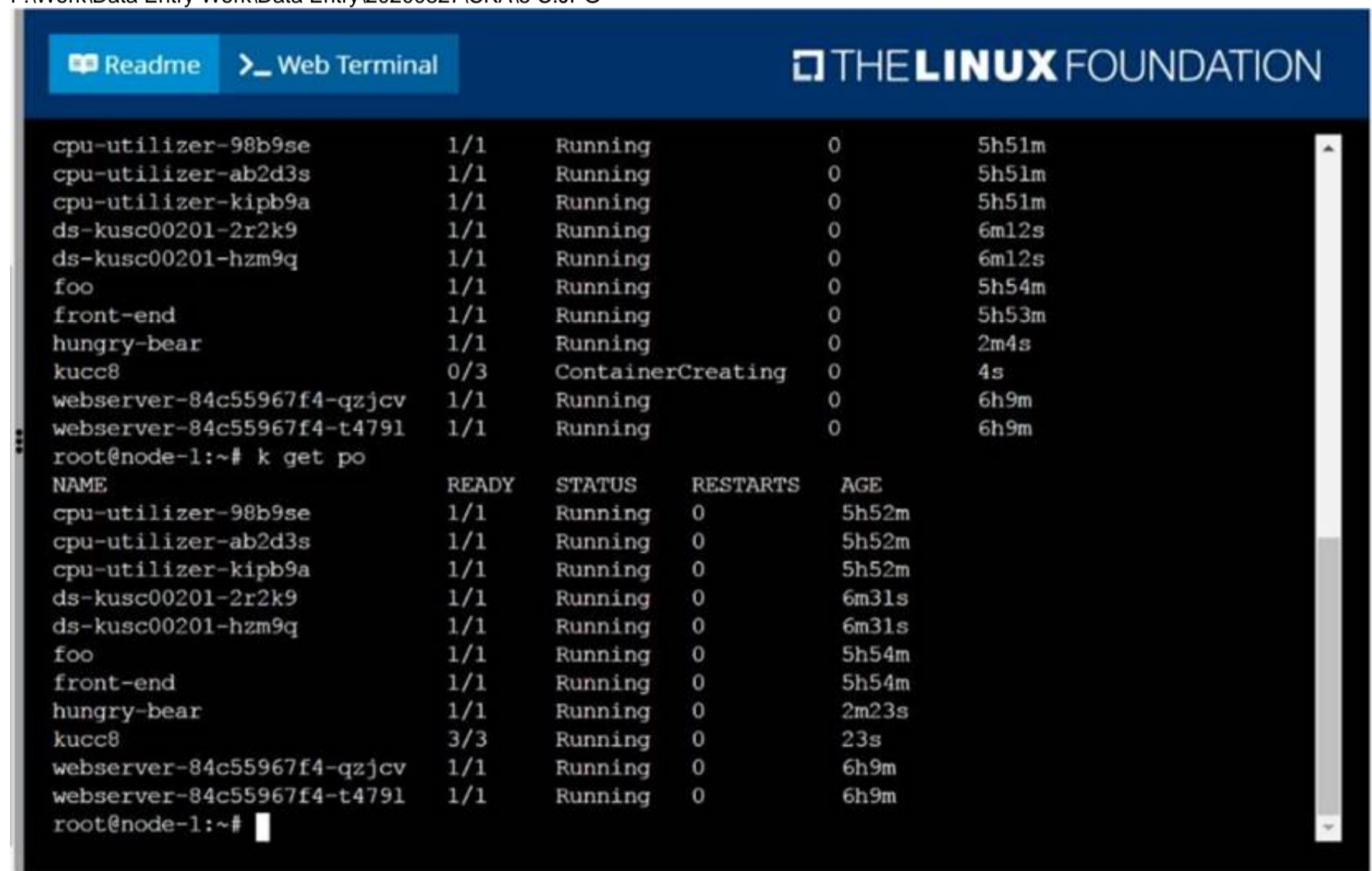
Explanation:**solution**

```
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:~# k get po
NAME                                READY   STATUS    RESTARTS   AGE
cpu-utilizer-98b9se                 1/1     Running   0           5h50m
cpu-utilizer-ab2d3s                 1/1     Running   0           5h50m
cpu-utilizer-kipb9a                 1/1     Running   0           5h50m
ds-kusc00201-2r2k9                  1/1     Running   0           4m50s
ds-kusc00201-hzm9q                  1/1     Running   0           4m50s
foo                                 1/1     Running   0           5h52m
front-end                           1/1     Running   0           5h52m
hungry-bear                         1/1     Running   0           42s
webserver-84c55967f4-qzjcv          1/1     Running   0           6h7m
webserver-84c55967f4-t479l          1/1     Running   0           6h7m
root@node-1:~# k run nginx --image=nginx --dry-run=client -o yaml > nginx.yaml
root@node-1:~# vim nginx.yaml
```

F:\Work\Data Entry Work\Data Entry\20200827\CKA\5 B.JPG



F:\Work\Data Entry Work\Data Entry\20200827\CKA\5 C.JPG



F:\Work\Data Entry Work\Data Entry\20200827\CKA\5 D.JPG

NEW QUESTION 22

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 25

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

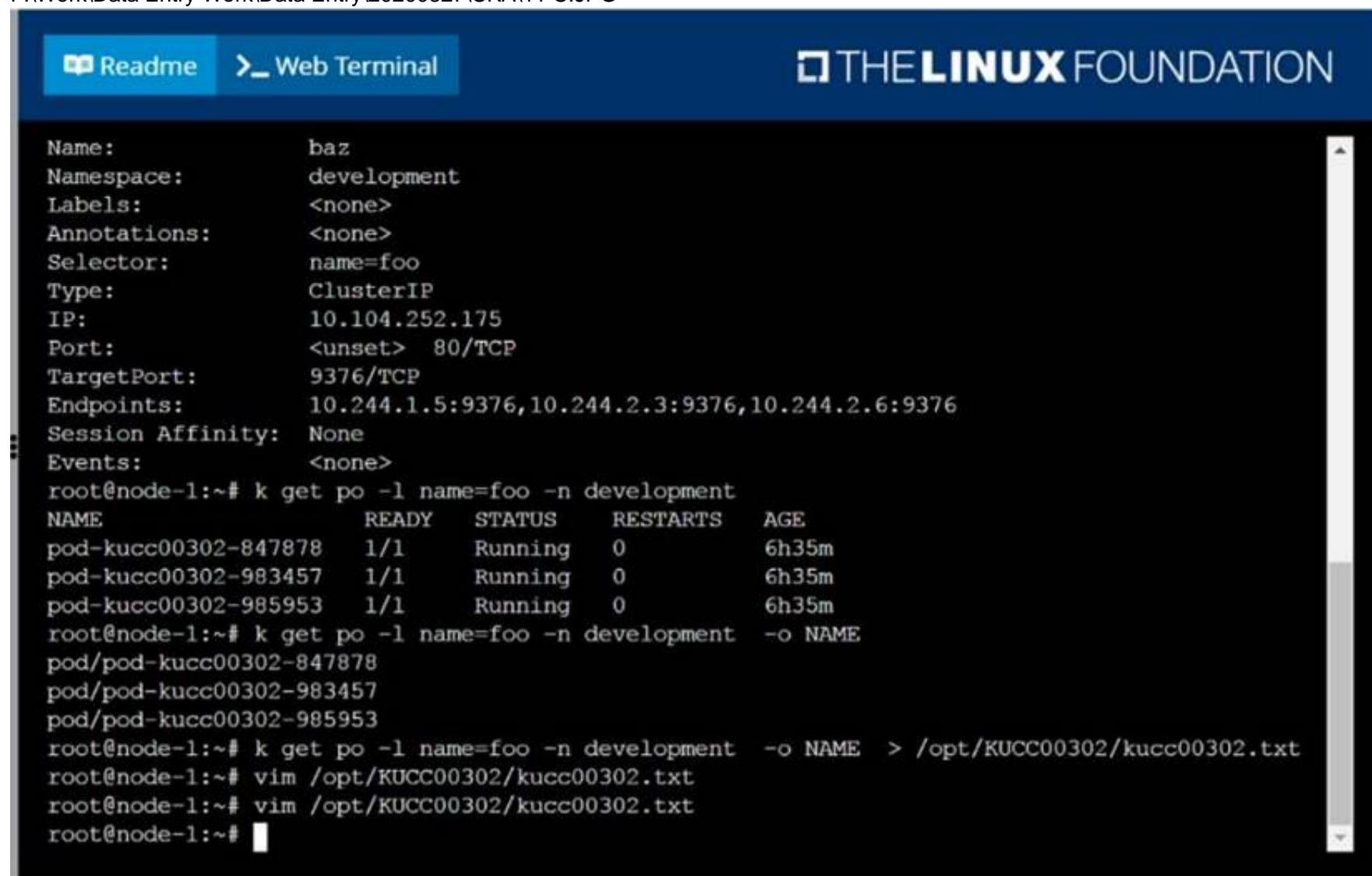
```
root@node-1:~# k describe svc baz -n development
Name:          baz
Namespace:     development
Labels:        <none>
Annotations:   <none>
Selector:      name=foo
Type:          ClusterIP
IP:            10.104.252.175
Port:          <unset> 80/TCP
TargetPort:    9376/TCP
Endpoints:     10.244.1.5:9376,10.244.2.3:9376,10.244.2.6:9376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

The image shows a web terminal interface. At the top, there is a dark blue header bar. On the left side of this bar, there are two buttons: 'Readme' with a book icon and 'Web Terminal' with a terminal icon. On the right side of the header bar is the 'THE LINUX FOUNDATION' logo. The main area of the terminal is black. It contains the following text: 'pod-kucc00302-847878', 'pod-kucc00302-983457', 'pod-kucc00302-985953', followed by a vertical line of approximately 20 tilde (~) characters. At the bottom left of the terminal area, there is a prompt ': WG' followed by a white cursor block. A vertical scrollbar is located on the right edge of the terminal area.

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



The screenshot shows a web terminal interface with a blue header containing "Readme" and "Web Terminal" tabs, and "THE LINUX FOUNDATION" logo. The terminal output displays the details of a pod named "baz" in the "development" namespace. The pod is a ClusterIP type with IP 10.104.252.175 and target port 9376/TCP. It has endpoints 10.244.1.5:9376, 10.244.2.3:9376, and 10.244.2.6:9376. The user then runs a command to get pods with label "name=foo" in the "development" namespace, which returns a table of three running pods. Finally, the user runs a command to get the pod names and save them to a file.

```
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
root@node-1:~# vim /opt/KUCC00302/kucc00302.txt
root@node-1:~#
```

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

NEW QUESTION 26

CORRECT TEXT

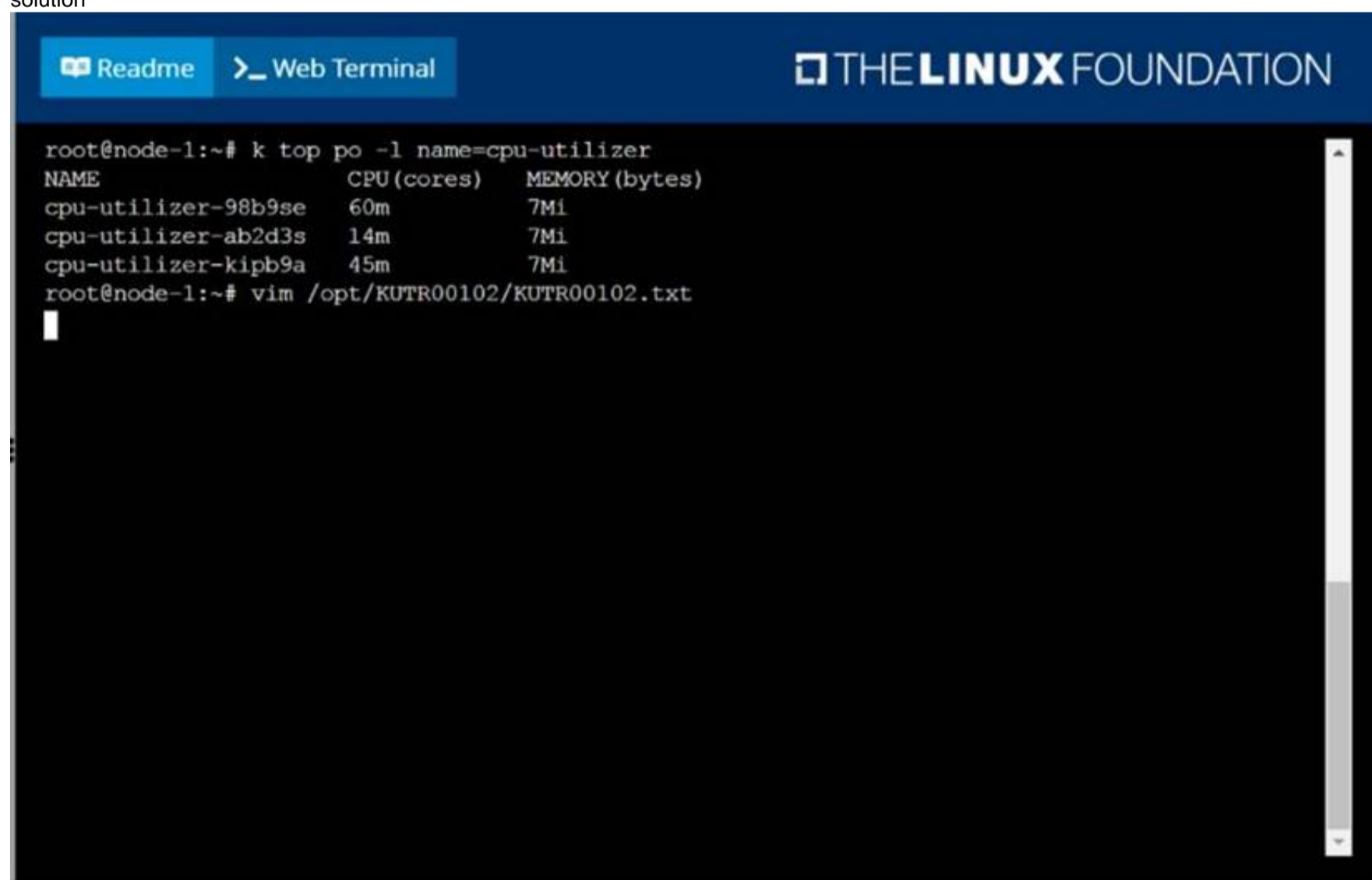
From the pod label name=cpu-utilizer, find pods running high CPU workloads and write the name of the pod consuming most CPU to the file /opt/KUTR00102/KUTR00102.txt (which already exists).

A.

Answer: Seethesolutionbelow.

Explanation:

solution

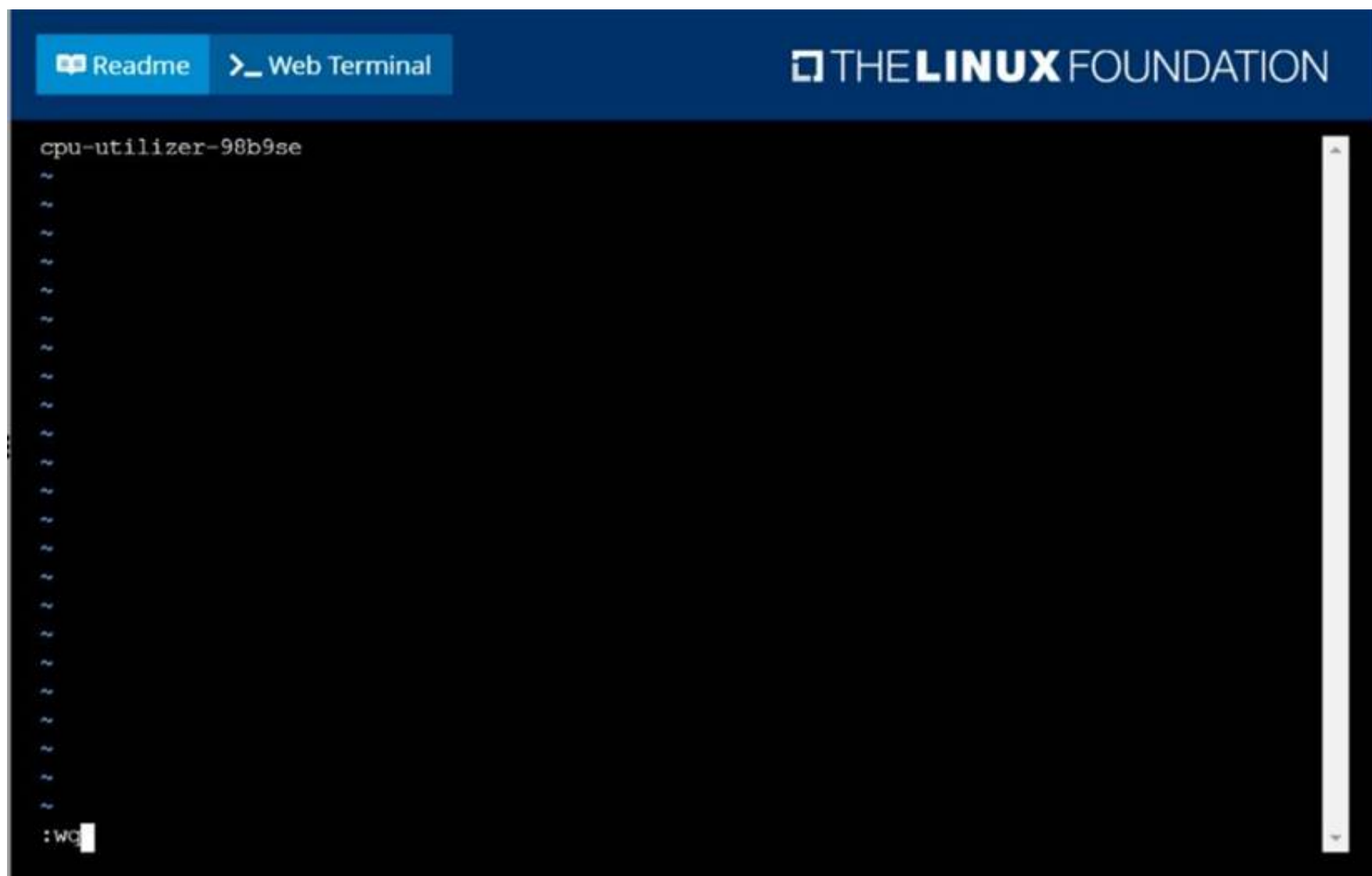


The screenshot shows a web terminal interface with a blue header containing "Readme" and "Web Terminal" tabs, and "THE LINUX FOUNDATION" logo. The terminal output shows the user running a command to get pods with label "name=cpu-utilizer" sorted by CPU usage. The output shows three pods: "cpu-utilizer-98b9se" (60m CPU), "cpu-utilizer-ab2d3s" (14m CPU), and "cpu-utilizer-kipb9a" (45m CPU). The user then runs a command to open a file in vim.

```
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
cpu-utilizer-kipb9a 45m 7Mi
root@node-1:~# vim /opt/KUTR00102/KUTR00102.txt

```

F:\Work\Data Entry Work\Data Entry\20200827\CKA\16 B.JPG



F:\Work\Data Entry Work\Data Entry\20200827\CKA\16 C.JPG

NEW QUESTION 29

CORRECT TEXT

Create a pod as follows:

? Name: mongo

? Using Image: mongo

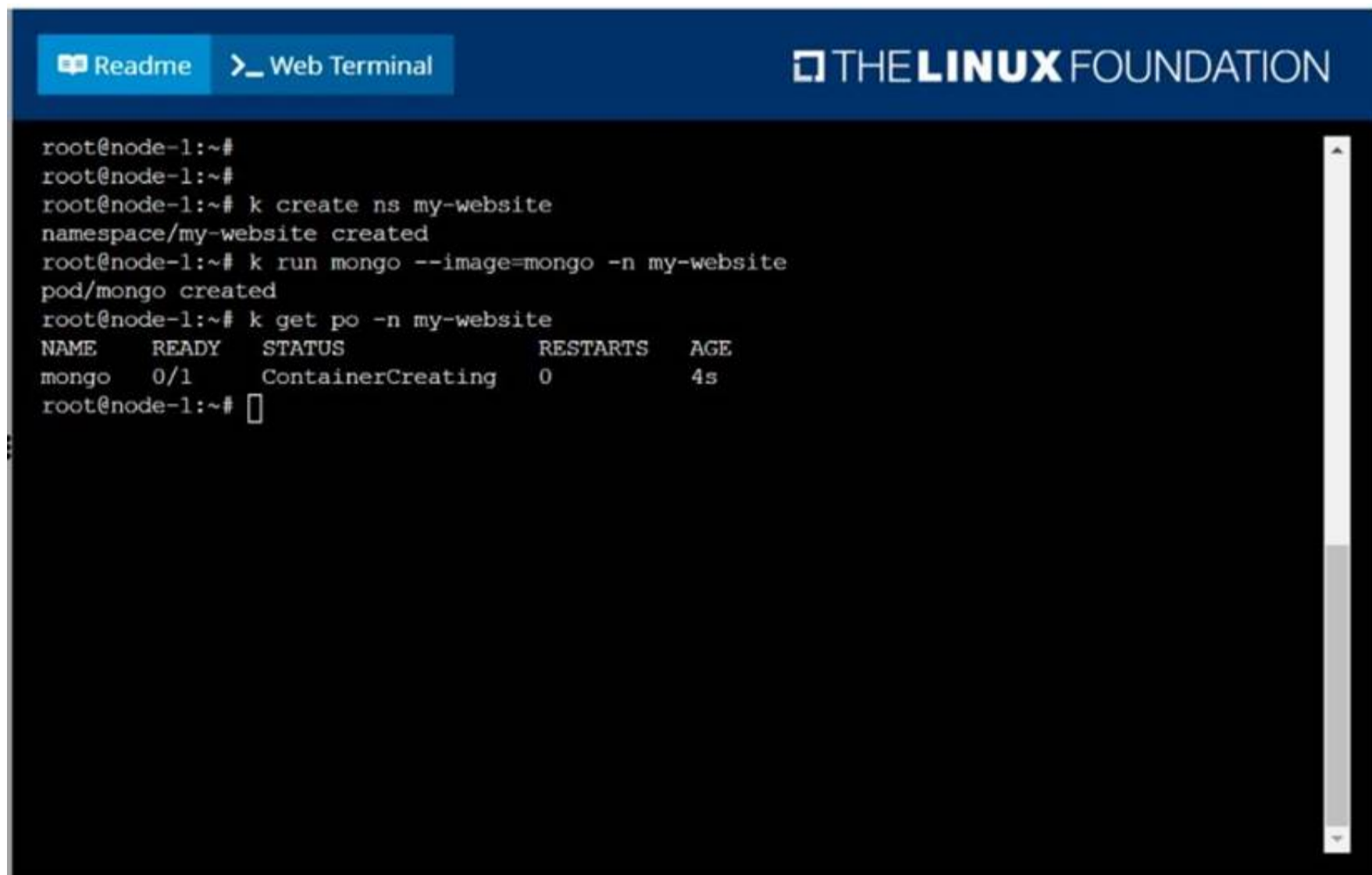
? In a new Kubernetes namespace named: my-website

- A. Mastered
- B. Not Mastered

Answer: A

Explanation:

solution



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

NEW QUESTION 31

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.

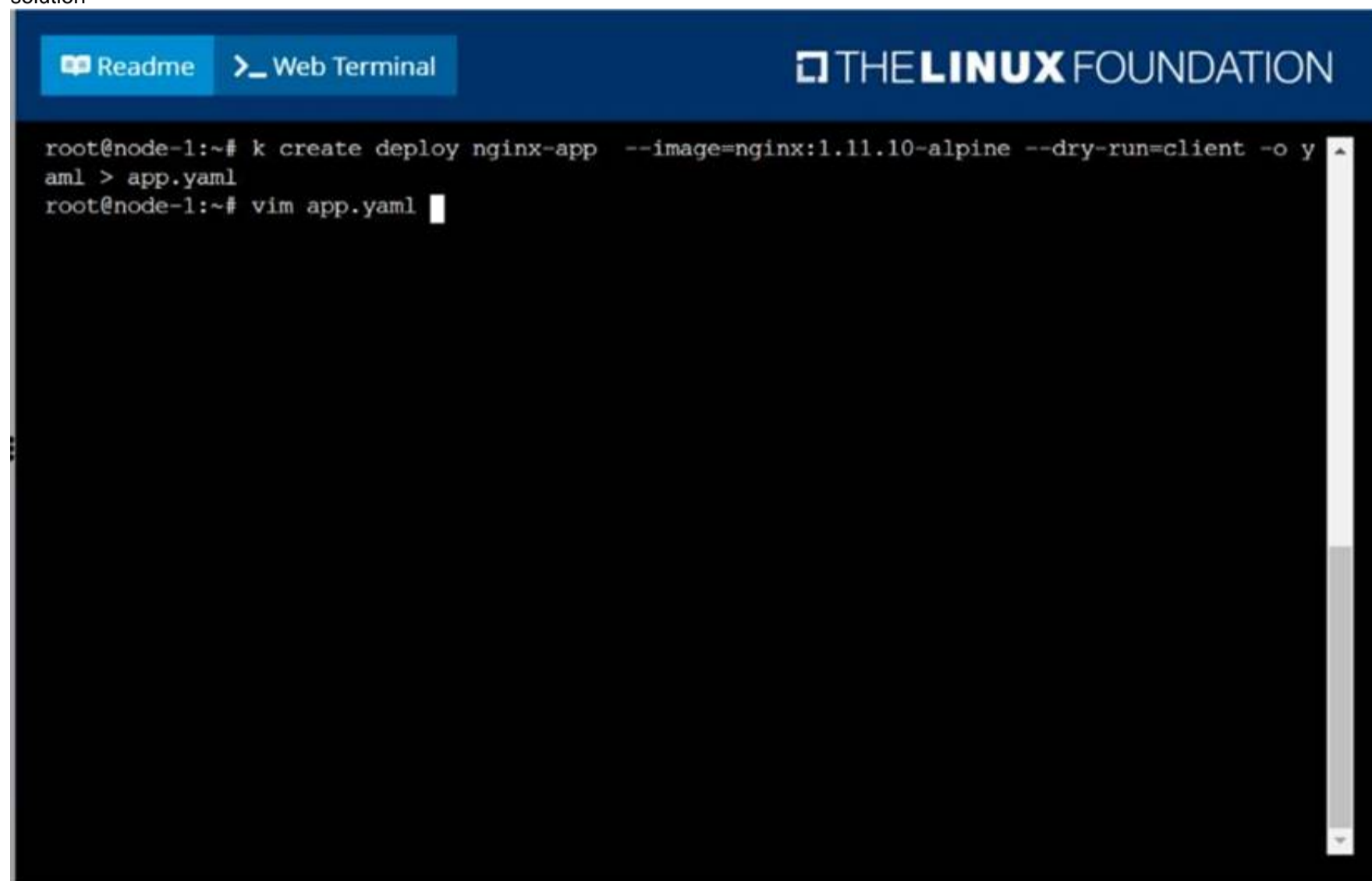
A. Mastered

B. Not Mastered

Answer: A

Explanation:

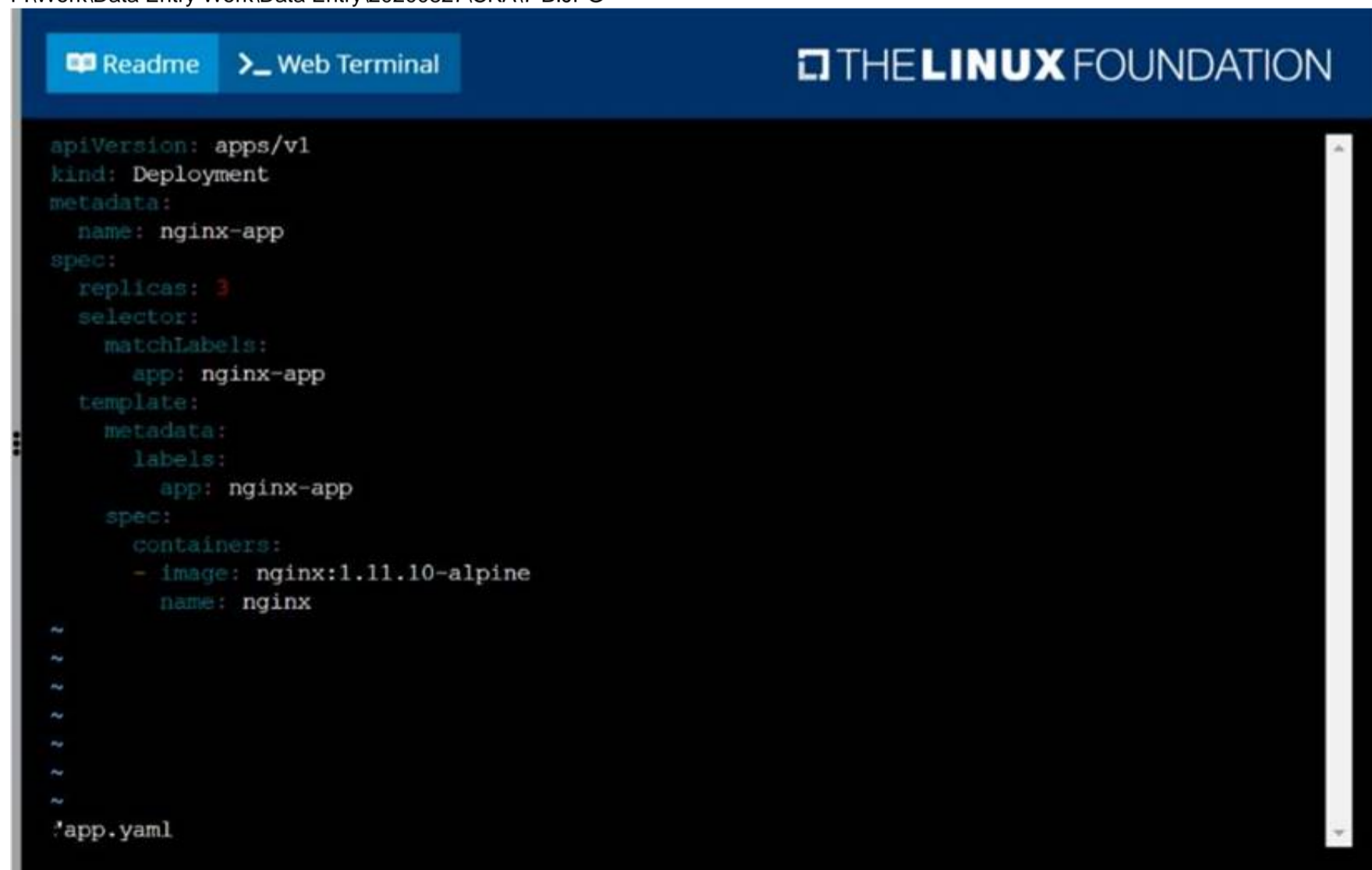
solution



The screenshot shows a web terminal interface with a dark background. At the top, there is a blue header bar with the text "THE LINUX FOUNDATION" on the right and two buttons: "Readme" and "Web Terminal". Below the header, the terminal 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
```

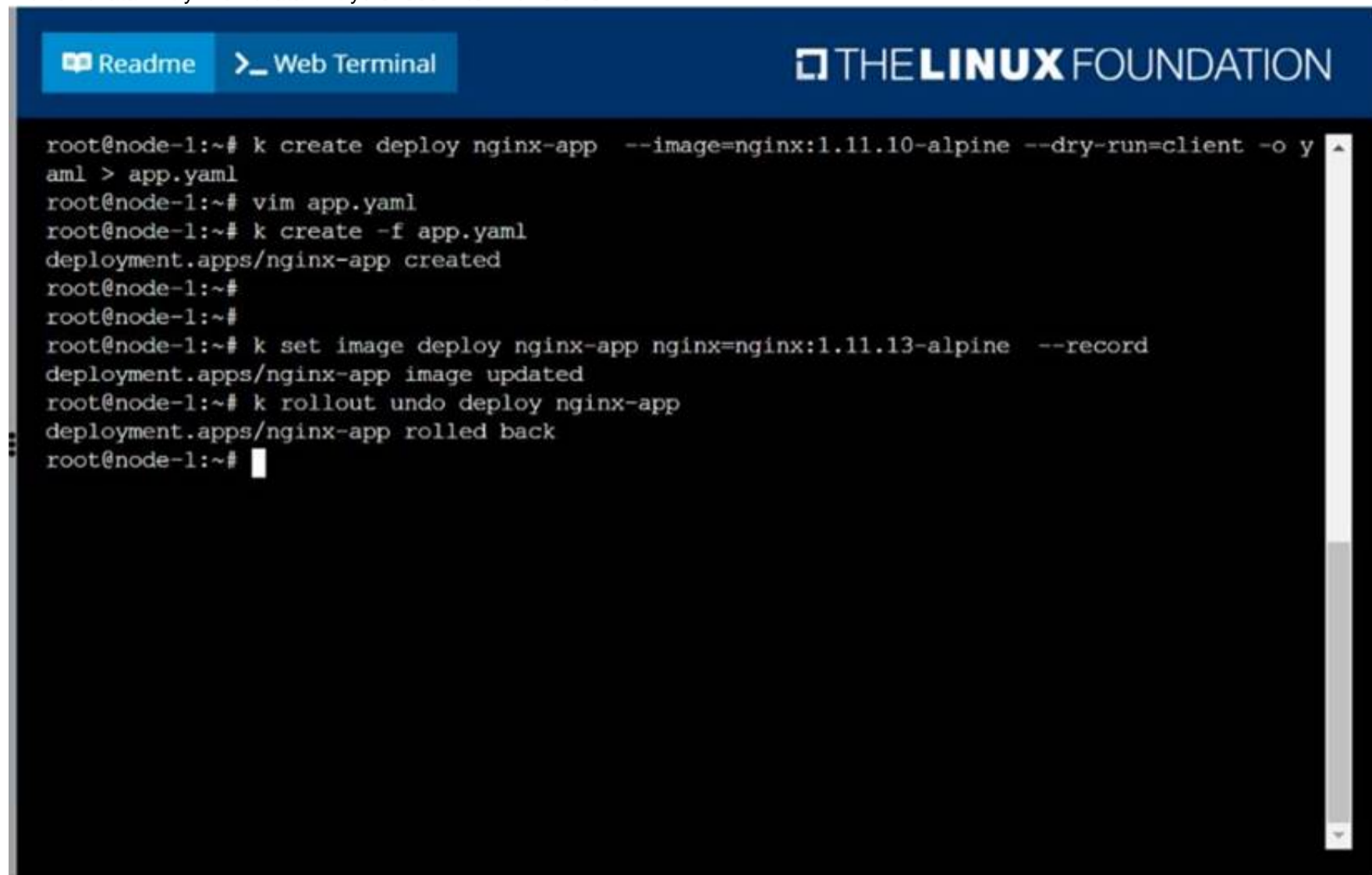
F:\Work\Data Entry Work\Data Entry\20200827\CKA\7 B.JPG



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

```
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
```

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



```
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:~#
```

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

NEW QUESTION 33

CORRECT TEXT

Create a pod that echo “hello world” and then exists. Have the pod deleted automatically when it’s completed

- A. Mastered
- B. Not Mastered

Answer: A**Explanation:**

```
kubectl run busybox --image=busybox -it --rm --restart=Never --
/bin/sh -c 'echo hello world'
kubectl get po # You shouldn't see pod with the name "busybox"
```

NEW QUESTION 36

CORRECT TEXT

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

- A. Mastered
- B. Not Mastered

Answer: A**Explanation:**

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

NEW QUESTION 41

CORRECT TEXT

Create a pod as follows:

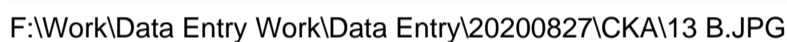
- ? Name: non-persistent-redis
- ? container Image: redis
- ? Volume with name: cache-control
- ? Mount path: /data/redis

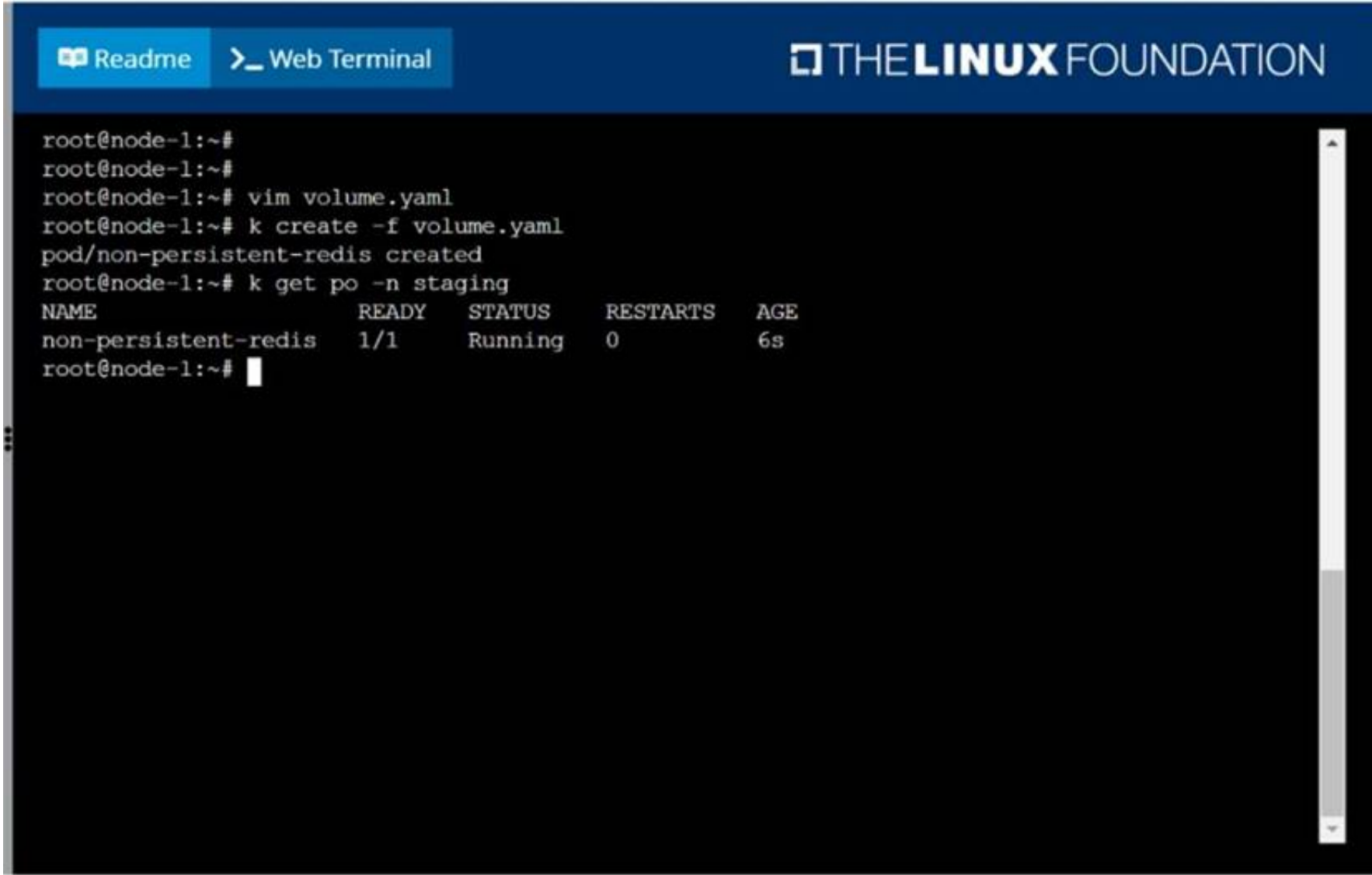
The pod should launch in the staging namespace and the volume must not be persistent.

- A. Mastered
- B. Not Mastered

Answer: A**Explanation:**

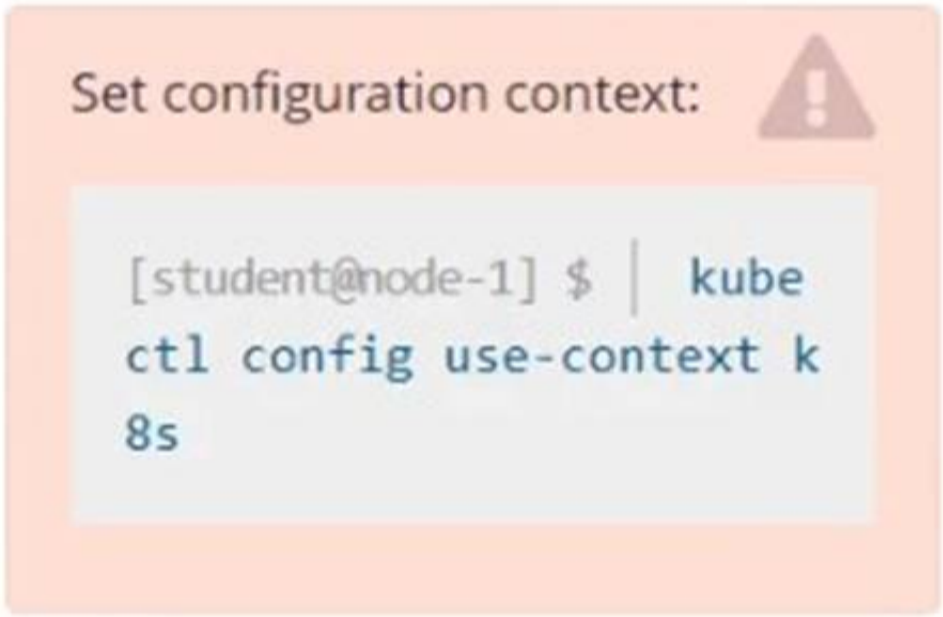
solution





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

NEW QUESTION 46
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 50

CORRECT TEXT

Task Weight: 4%



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 51

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 54

CORRECT TEXT

Perform the following tasks:

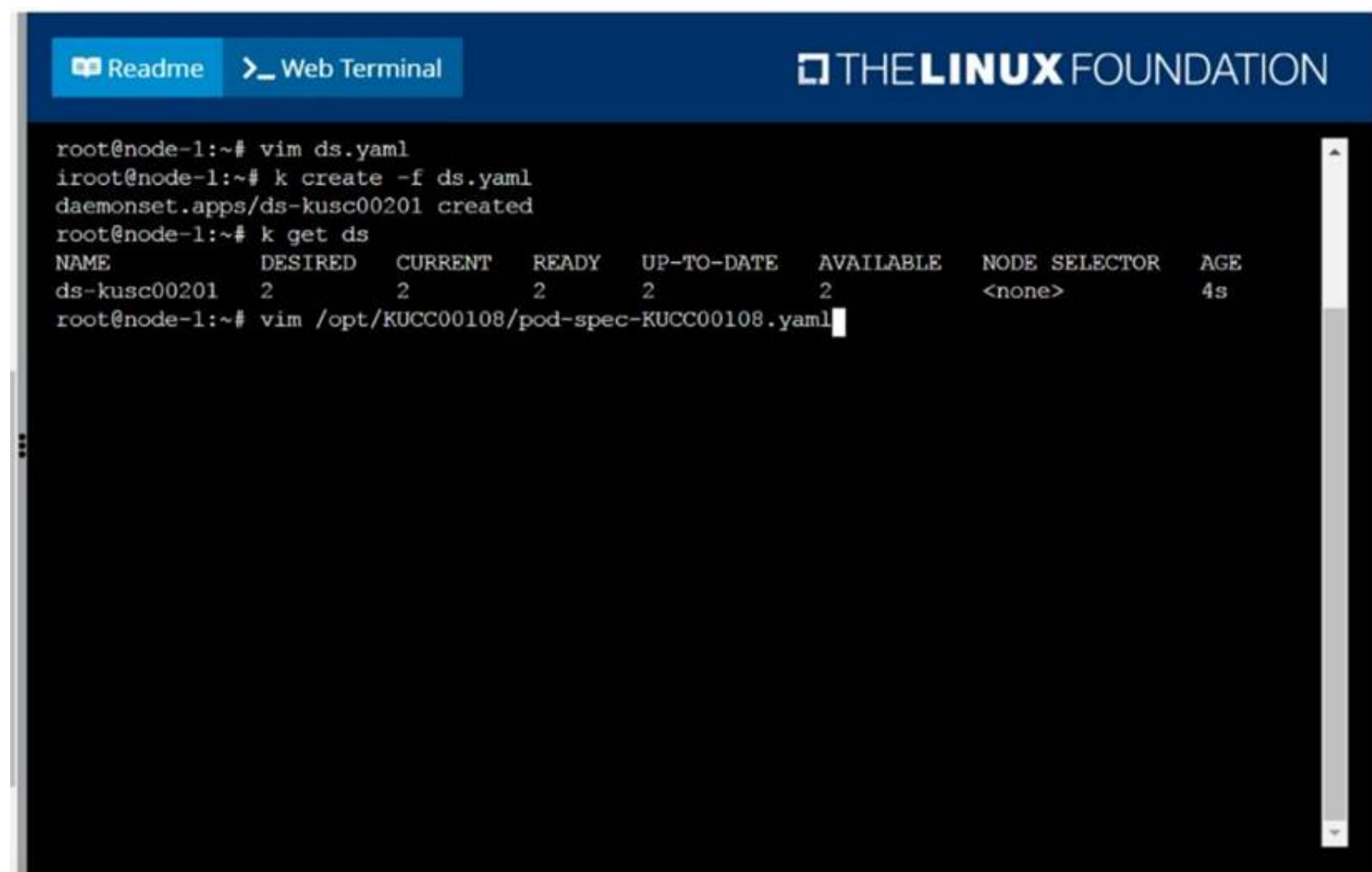
- ? Add an init container to hungry-bear (which has been defined in spec file /opt/KUCC00108/pod-spec-KUCC00108.yaml)
- ? The init container should create an empty file named/workdir/calm.txt
- ? If /workdir/calm.txt is not detected, the pod should exit
- ? Once the spec file has been updated with the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

A.

Answer: Seethesolutionbelow.

Explanation:

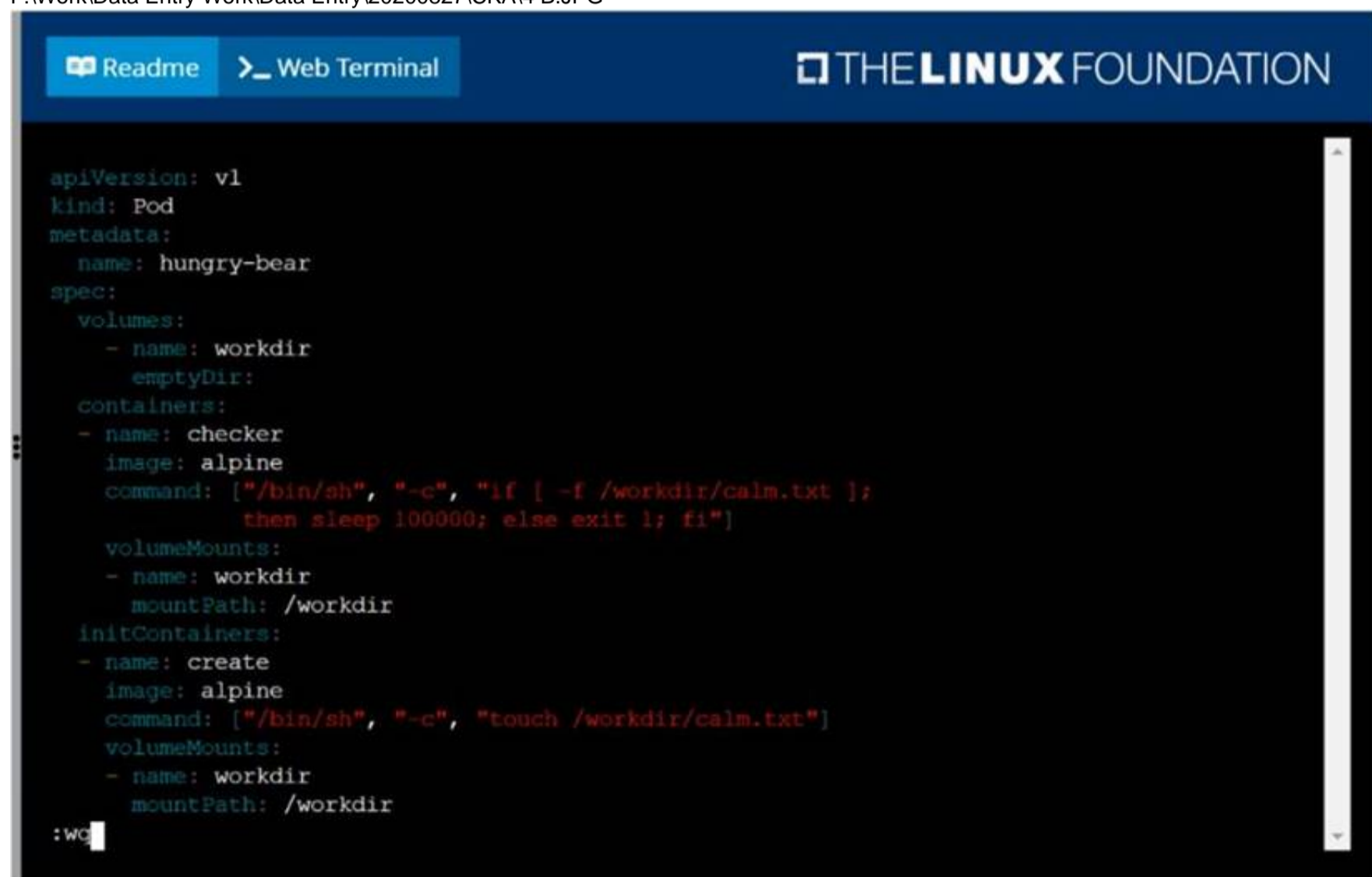
solution



The screenshot shows a web terminal interface with a dark background. At the top, there is a blue header bar with the text "THE LINUX FOUNDATION" on the right. On the left of the header, there are two buttons: "Readme" and "Web Terminal". The terminal content shows a series of commands and their outputs:

```
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
```

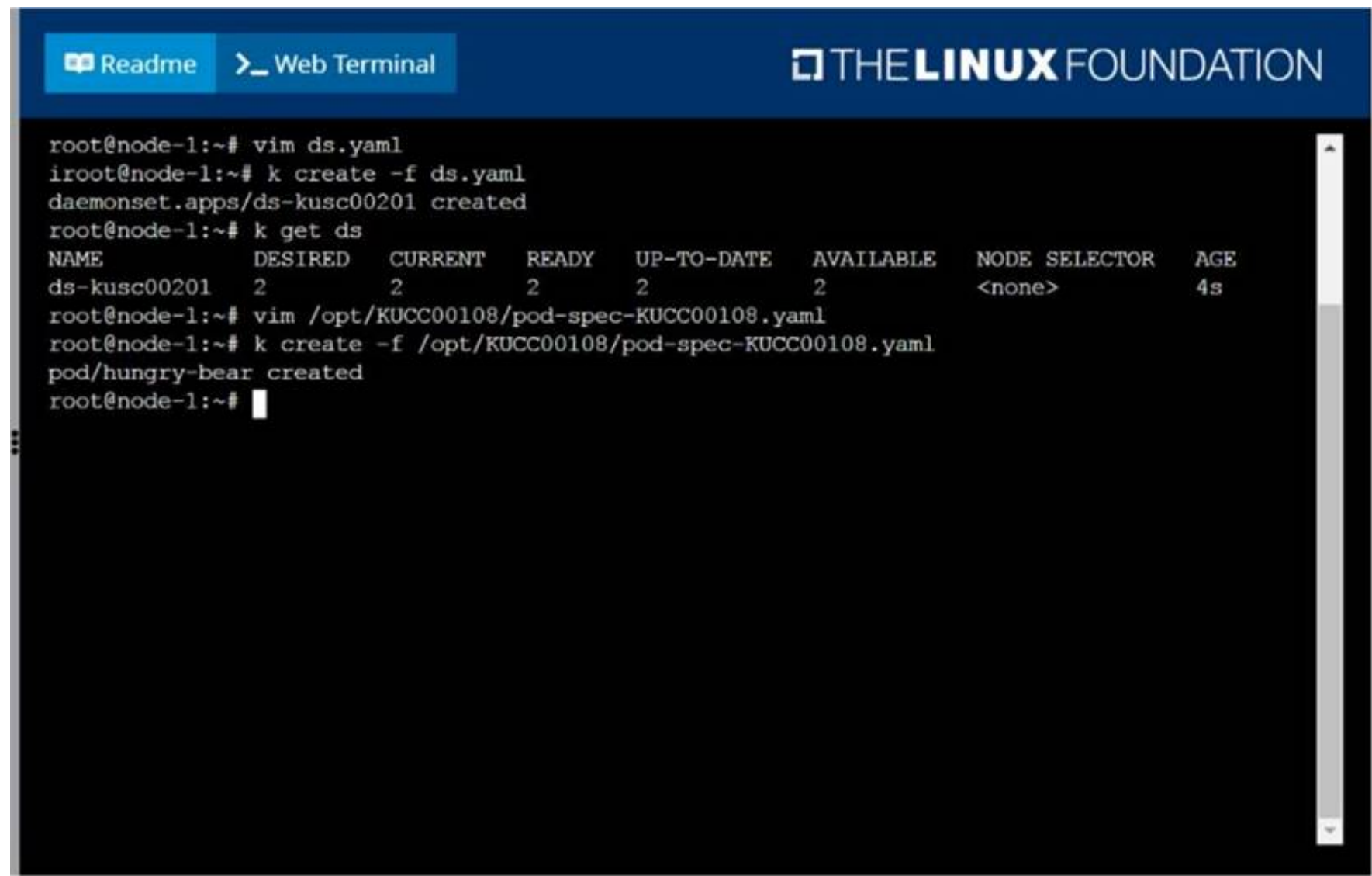
F:\Work\Data Entry Work\Data Entry\20200827\CKA\4 B.JPG



The screenshot shows a web terminal interface with a dark background. At the top, there is a blue header bar with the text "THE LINUX FOUNDATION" on the right. On the left of the header, there are two buttons: "Readme" and "Web Terminal". The terminal content shows a Kubernetes 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
: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 57

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:

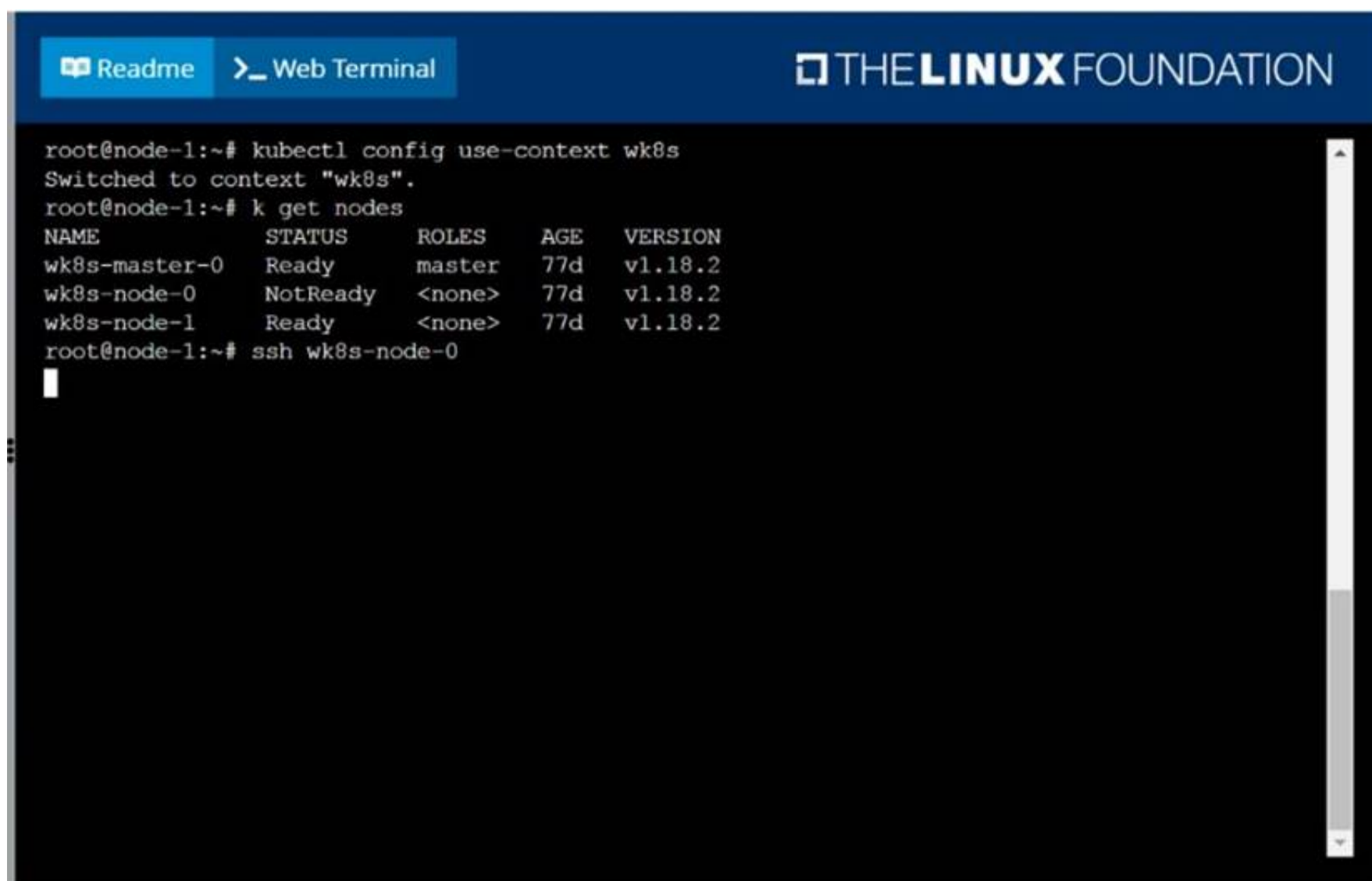
[student@w8ks-node-0] \$ | sudo -i

- A. Mastered
- B. Not Mastered

Answer: A

Explanation:

solution

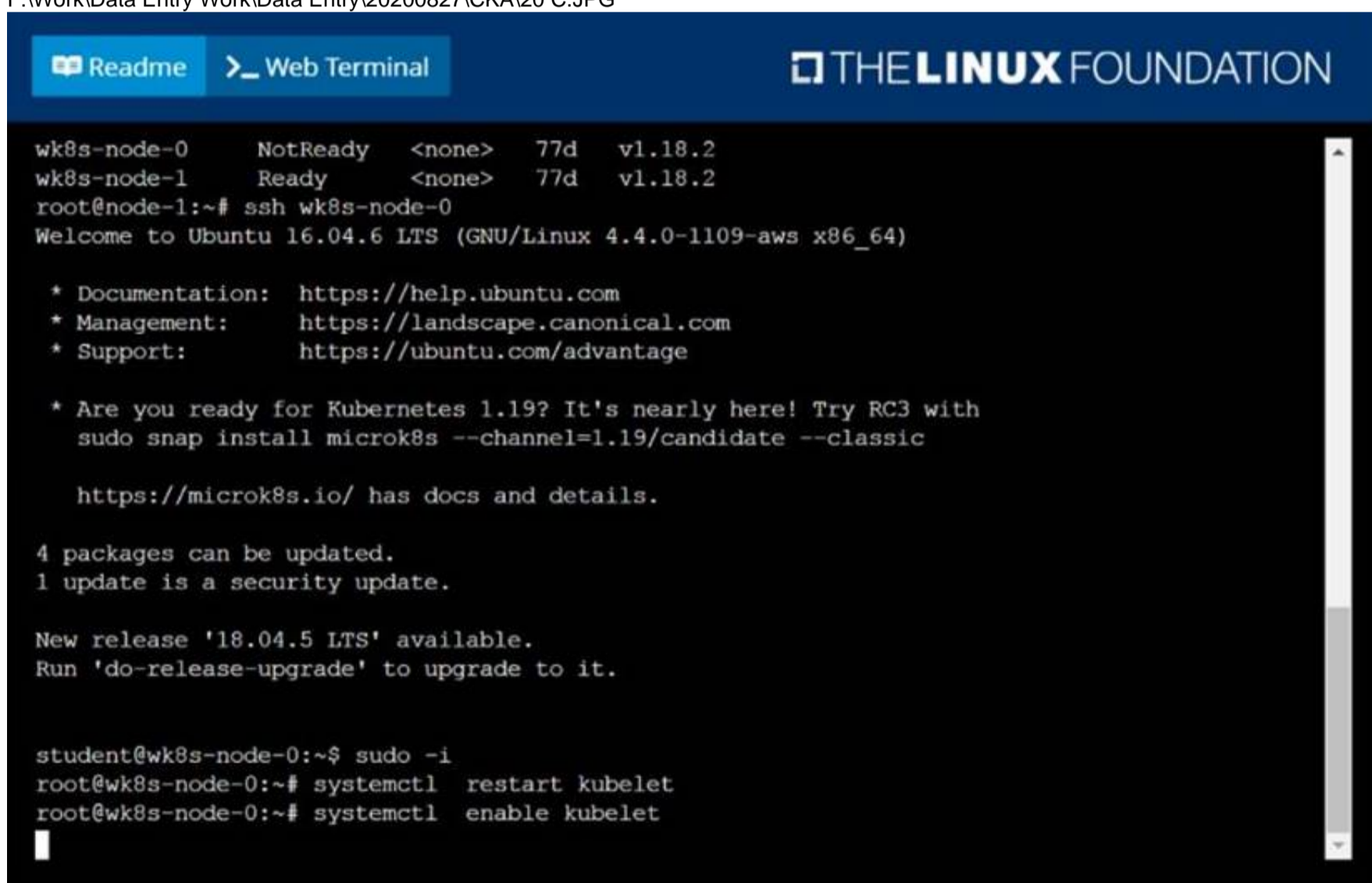


The screenshot shows a web terminal interface with a dark blue header. On the left, there are two tabs: 'Readme' and 'Web Terminal'. On the right, the 'THE LINUX FOUNDATION' logo is displayed. The terminal content shows a user at 'root@node-1' running 'kubectl config use-context wk8s', which switches the context to 'wk8s'. Then, the user runs 'k get nodes', displaying a table of node statuses. Finally, the user runs 'ssh wk8s-node-0', and the terminal shows a blank line, indicating the connection is established.

```
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



The screenshot shows a web terminal interface with a dark blue header. On the left, there are two tabs: 'Readme' and 'Web Terminal'. On the right, the 'THE LINUX FOUNDATION' logo is displayed. The terminal content shows a user at 'root@node-1' running 'ssh wk8s-node-0'. The terminal then shows the Ubuntu 16.04.6 LTS login screen with various links for documentation, management, and support. It also displays system update information, including a security update and a new release of Ubuntu. Finally, the user runs 'sudo -i' to become 'student@wk8s-node-0', and then runs 'systemctl restart kubelet' and 'systemctl enable kubelet' to restart and enable the kubelet service.

```
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.

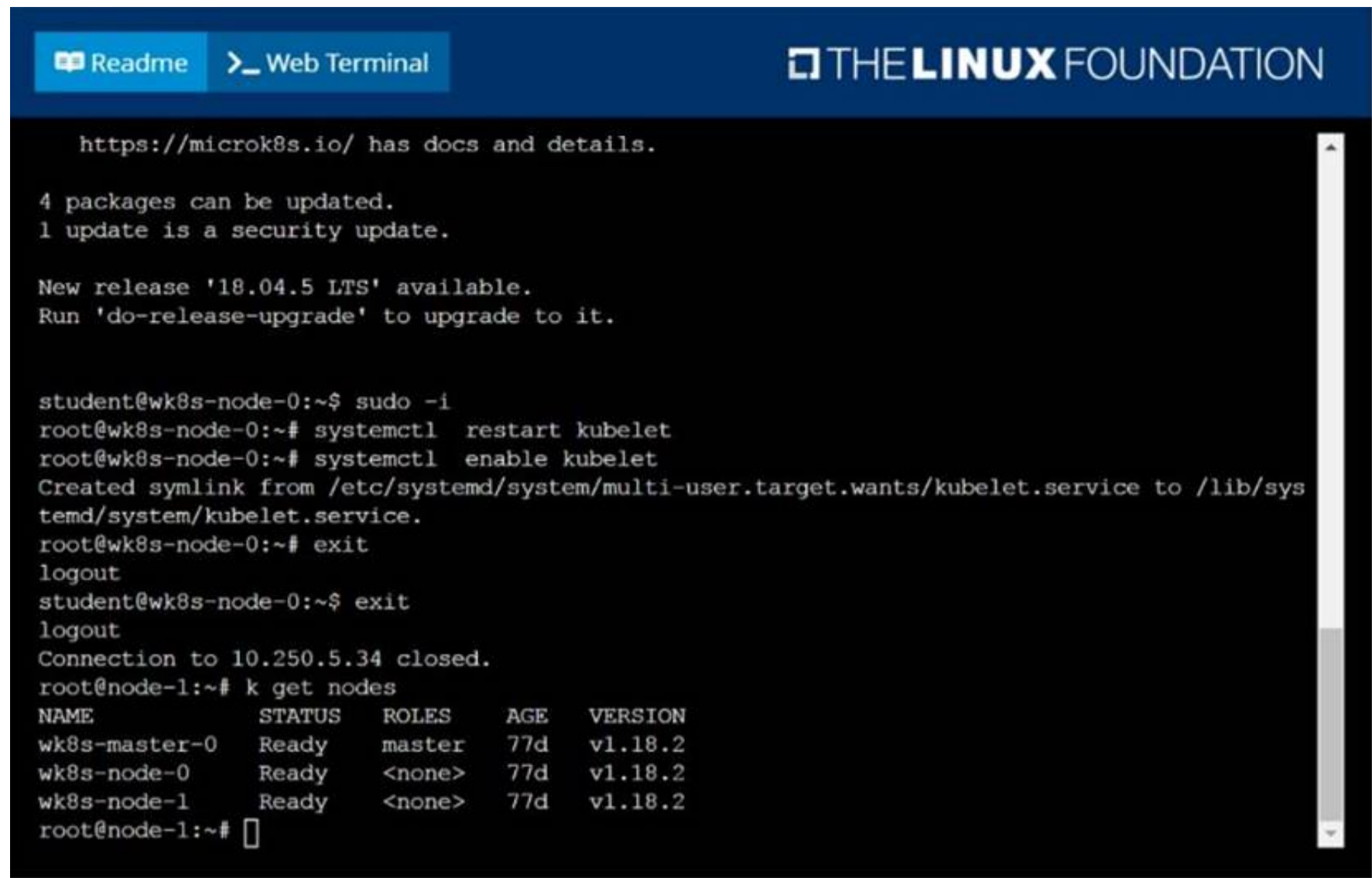
4 packages can be updated.
1 update is a security update.

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



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

NEW QUESTION 61

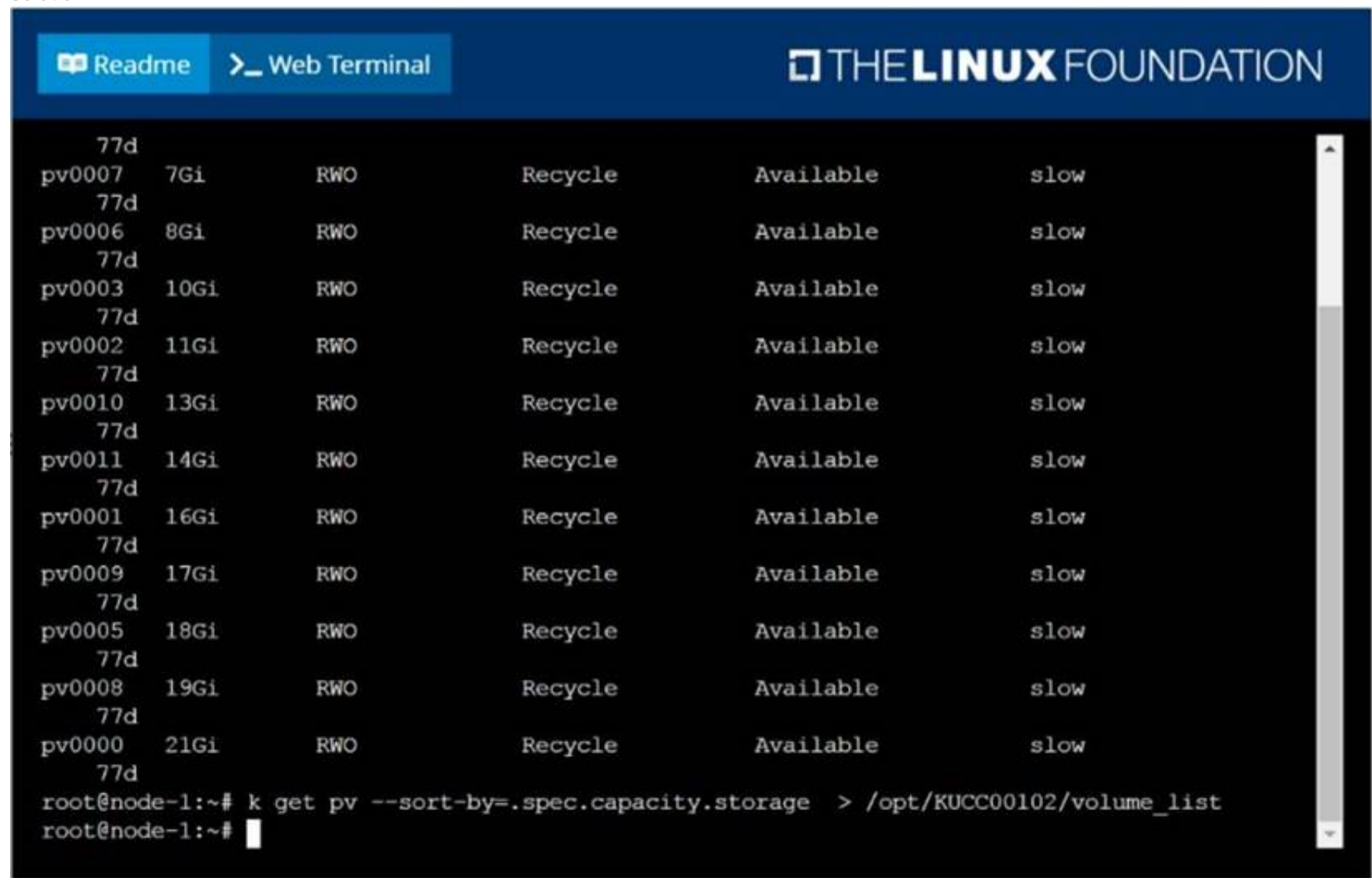
CORRECT TEXT

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 63

CORRECT TEXT

Create a busybox pod and add “sleep 3600” command

- A. Mastered
B. Not Mastered

Answer: A**Explanation:**

```
kubectl run busybox --image=busybox --restart=Never -- /bin/sh -c  
"sleep 3600"
```

NEW QUESTION 66

CORRECT TEXT

Score: 5%



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 71

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 73

CORRECT TEXT

Score: 4%



Task

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:

```
[student@node-1] > ssh ek8s
```

```
kubectl cordon ek8s-node-1
```

```
kubectl drain ek8s-node-1 --delete-local-data --ignore-daemonsets --force
```

NEW QUESTION 77

.....

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>