

# Linux-Foundation

## Exam Questions CKA

Certified Kubernetes Administrator (CKA) Program



NEW QUESTION 1

CORRECT TEXT

List all the pods sorted by name

- A. Mastered
- B. Not Mastered

Answer: A

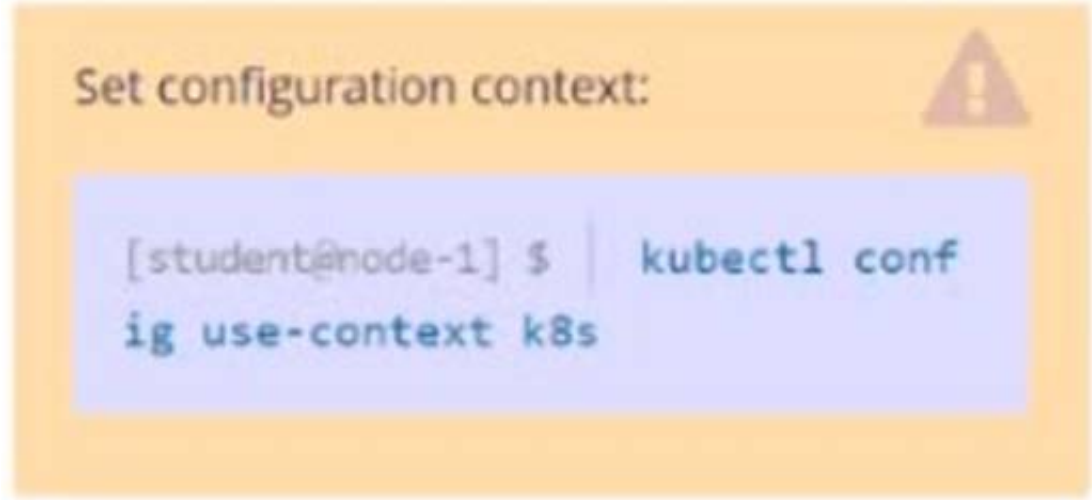
Explanation:

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

NEW QUESTION 2

CORRECT TEXT

Task Weight: 4%



Task

Schedule a Pod as follows:

- Name: kucc1
- App Containers: 2
- Container Name/Images: o nginx  
o consul

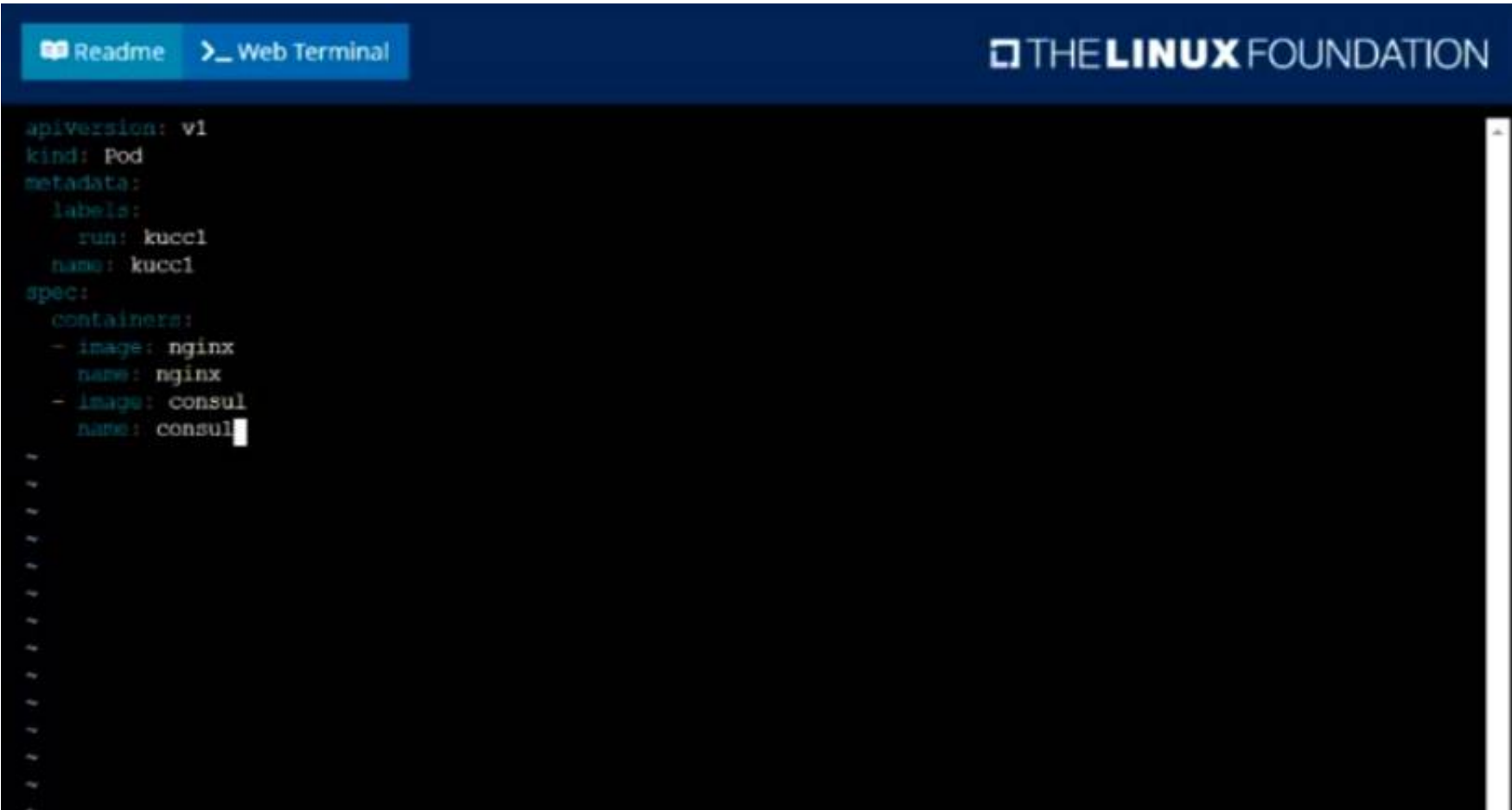
- A. Mastered
- B. Not Mastered

Answer: A

Explanation:

Solution:

```
student@node-1:~$ kubect1 config use-context k8s
Switched to context "k8s".
student@node-1:~$ kubect1 run kucc1 --image=nginx --dry-run=client -o yaml > aa.y
```



Graphical user interface, text, application  
Description automatically generated

```
student@node-1:~$ kubectl config use-context k8s
Switched to context "k8s".
student@node-1:~$ kubectl run kucc1 --image=nginx --dry-run=client -o yaml > aa.yaml
student@node-1:~$ vim aa.yaml
student@node-1:~$ kubectl create -f aa.yaml
pod/kucc1 created
student@node-1:~$ kubectl get pods
NAME                                READY   STATUS              RESTARTS   AGE
ll-factor-app                       1/1     Running             0           6h34m
cpu-loader-98b9se                   1/1     Running             0           6h33m
cpu-loader-ab2d3s                   1/1     Running             0           6h33m
cpu-loader-kipb9a                   1/1     Running             0           6h33m
foobar                              1/1     Running             0           6h34m
front-end-6bc87b9748-24rcm          1/1     Running             0           5m4s
front-end-6bc87b9748-hd5wp          1/1     Running             0           5m2s
kucc1                                0/2     ContainerCreating   0           3s
nginx-kusc00401                     1/1     Running             0           2m28s
webserver-84c89dfd75-2d1jn          1/1     Running             0           6h38m
webserver-84c89dfd75-8d8x2          1/1     Running             0           6h38m
webserver-84c89dfd75-z5zz4          1/1     Running             0           3m51s
student@node-1:~$
```

Text Description automatically generated

NEW QUESTION 3

CORRECT TEXT

Create a namespace called 'development' and a pod with image nginx called nginx on this namespace.

- A. Mastered
- B. Not Mastered

Answer: A

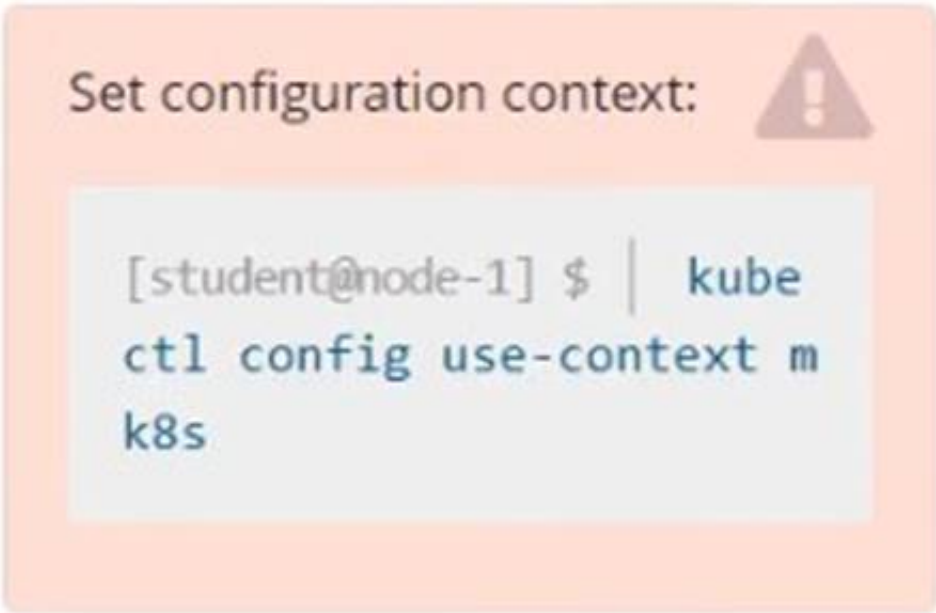
Explanation:

kubectl create namespace development  
kubectl run nginx --image=nginx --restart=Never -n development

NEW QUESTION 4

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 can ssh to the master node using:

```
[student@node-1] $ | ssh
mk8s-master-0
```

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

```
[student@mk8s-master-0] |
$
sudo -i
```

You are also expected to upgrade kubelet and kubectl 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 5**

CORRECT TEXT

Score: 4%



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 6

CORRECT TEXT

Scale the deployment webserver to 6 pods.

- A. Mastered
- B. Not Mastered


**Answer:** A

**Explanation:**

solution



Readme
Web Terminal



```

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 7

##### 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: `Topt/KUCM00302/etcd-client.key`


- A. Mastered
- B. Not Mastered

**Answer:** A

##### Explanation:

solution

Readme
Web Terminal



```

root@node-1:~# ETCDCTL_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":"created 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 8**

CORRECT TEXT

Check the image version in pod without the describe command

- A. Mastered
- B. Not Mastered

**Answer:** A**Explanation:**

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

**NEW QUESTION 9**

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 10**

CORRECT TEXT

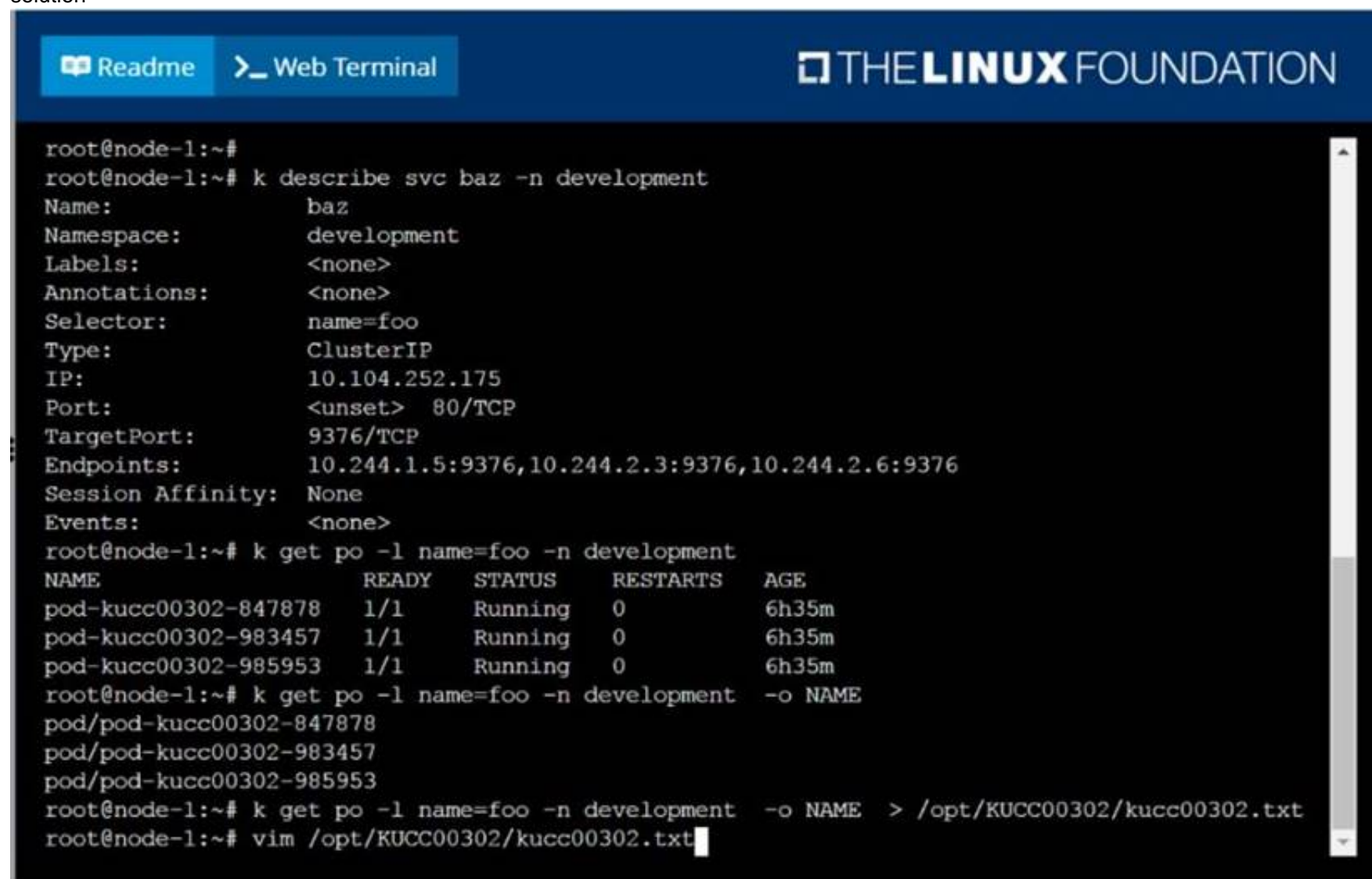
Create a file:

/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

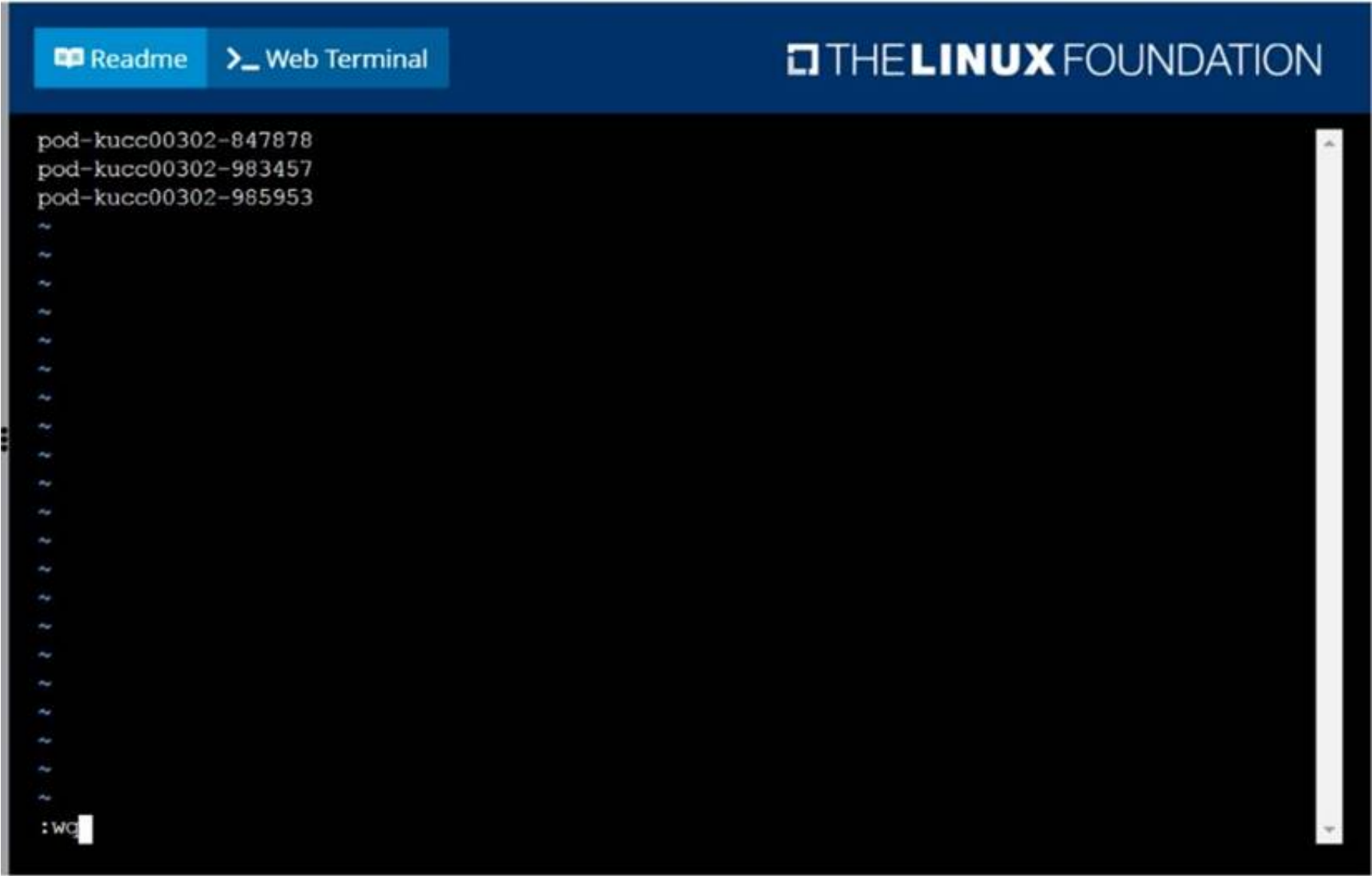
**Answer:** A**Explanation:**

solution

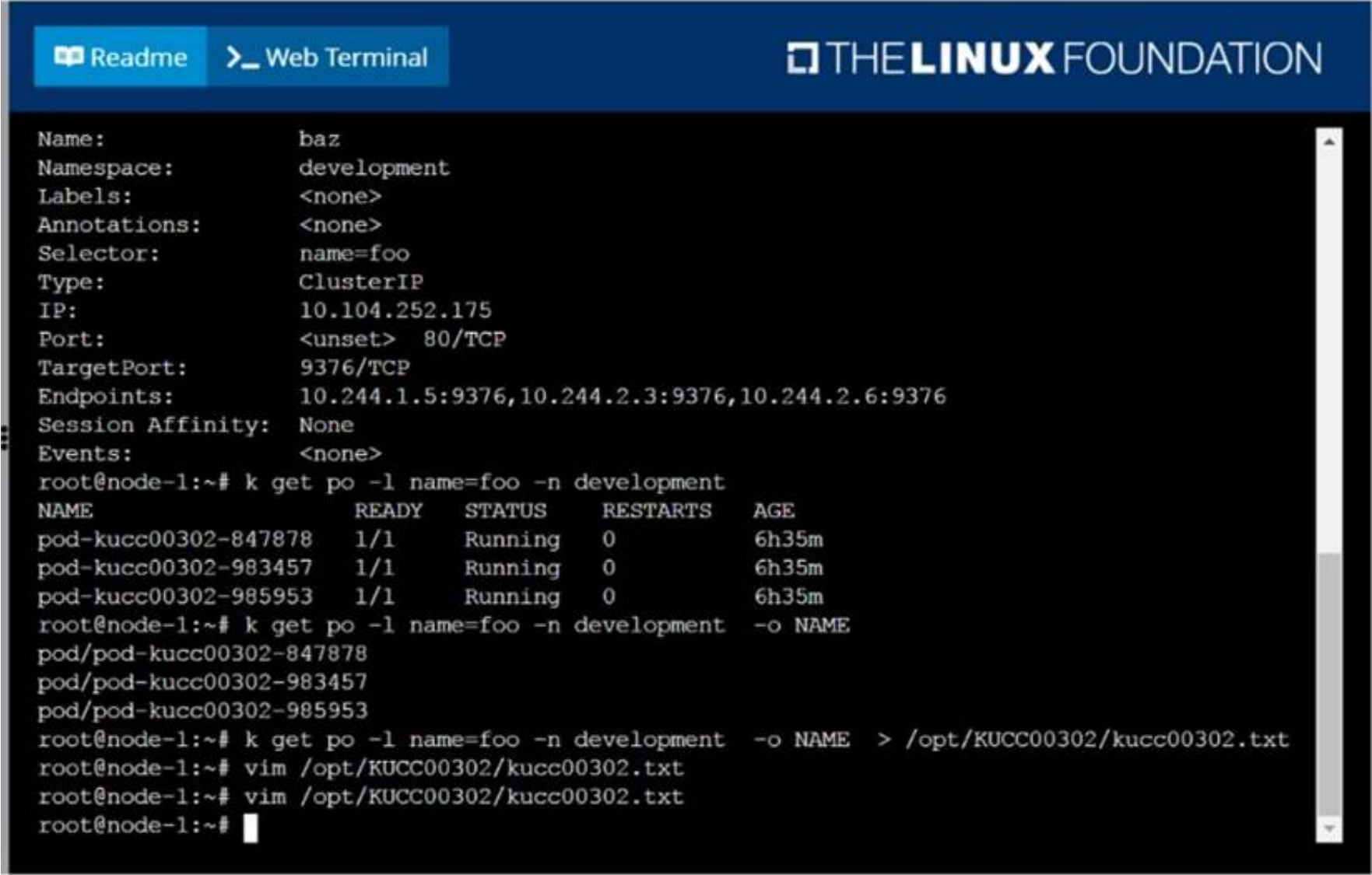


```
root@node-1:~#
root@node-1:~# k describe svc baz -n development
Name:          baz
Namespace:     development
Labels:        <none>
Annotations:   <none>
Selector:      name=foo
Type:          ClusterIP
IP:            10.104.252.175
Port:          <unset> 80/TCP
TargetPort:    9376/TCP
Endpoints: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



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



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

NEW QUESTION 10  
CORRECT TEXT  
Score: 4%



Set configuration context:



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

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 11

CORRECT TEXT

Score: 4%

Set configuration context:



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

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 14

##### CORRECT TEXT

Configure the kubelet systemd- managed service, on the node labelled with name=wk8s- node-1, to launch a pod containing a single container of Image httpd named webtool automatically. Any spec files required should be placed in the /etc/kubernetes/manifests directory on the node.

You can ssh to the appropriate node using:

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

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

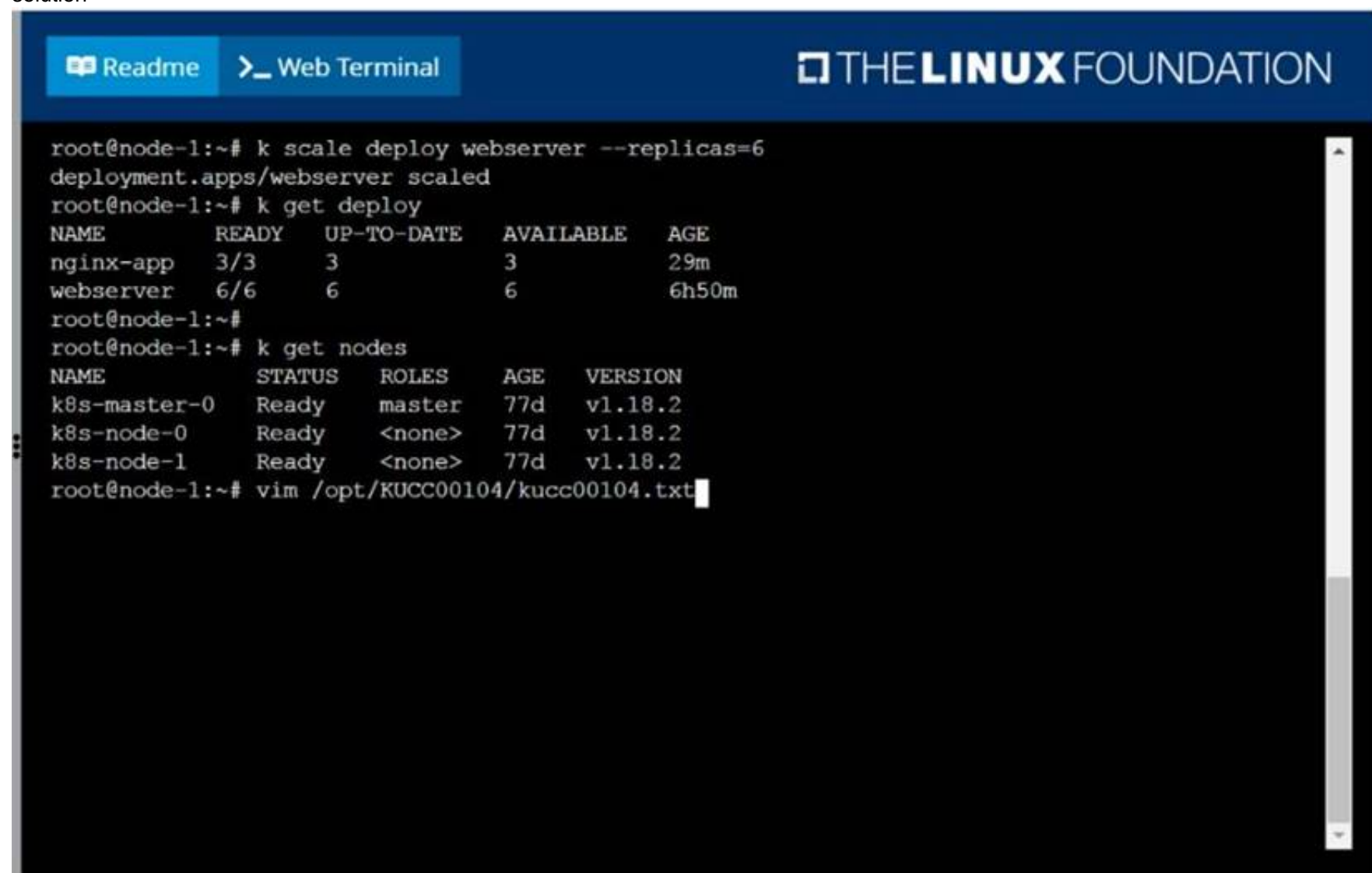
```
[student@wk8s-node-1] $ | sudo -i
```

- A. Mastered
- B. Not Mastered

**Answer:** A

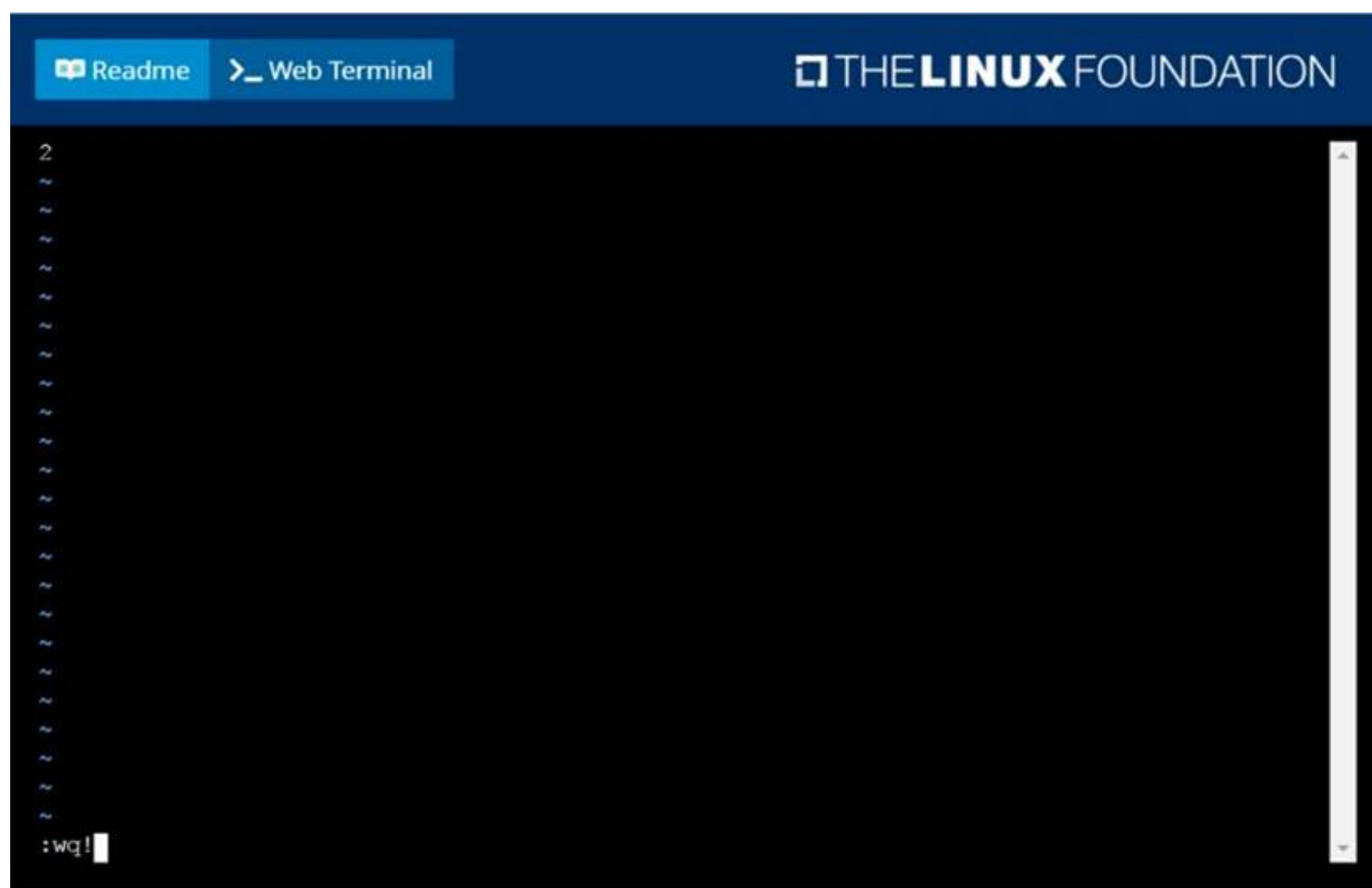
##### Explanation:

solution



```
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:~#
root@node-1:~# k get nodes
NAME           STATUS    ROLES    AGE   VERSION
k8s-master-0   Ready     master   77d   v1.18.2
k8s-node-0     Ready     <none>    77d   v1.18.2
k8s-node-1     Ready     <none>    77d   v1.18.2
root@node-1:~# vim /opt/KUCC00104/kucc00104.txt
```

F:\Work\Data Entry Work\Data Entry\20200827\CKA\15 B.JPG



F:\Work\Data Entry Work\Data Entry\20200827\CKA\15 C.JPG

#### NEW QUESTION 19

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 trying 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
```

\* 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)$ kubectl 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: v1 kind: Pod metadata: creationTimestamp: null name: app-data spec: volumes: - name: configpvc persistentVolumeClaim: claimName: app-data containers: - image: nginx name: app volumeMounts: - mountPath: "/srv/app-data" name: configpvc
```

## NEW QUESTION 21

NEW QUESTION  
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:**

```
kuibectl run busybox --image=busybox --restart=Never --rm -it -- env > envpod.yaml
```

**NEW QUESTION 25**



CORRECT TEXT

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

- A. Mastered
- B. Not Mastered

**Answer:** A

**Explanation:**

```
kubect! run nginx --image=nginx --restart=Never --labels=env=test
-- namespace=engineering --dry-run -o yaml > nginx-pod.yaml
kubect! run nginx --image=nginx --restart=Never --labels=env=test --
namespace=engineering --dry-run -o yaml | kubect! 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
kubect! create -f nginx-pod.yaml
```

**NEW QUESTION 30**

.....

## Thank You for Trying Our Product

### We offer two products:

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

2nd - Questons 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](#)**