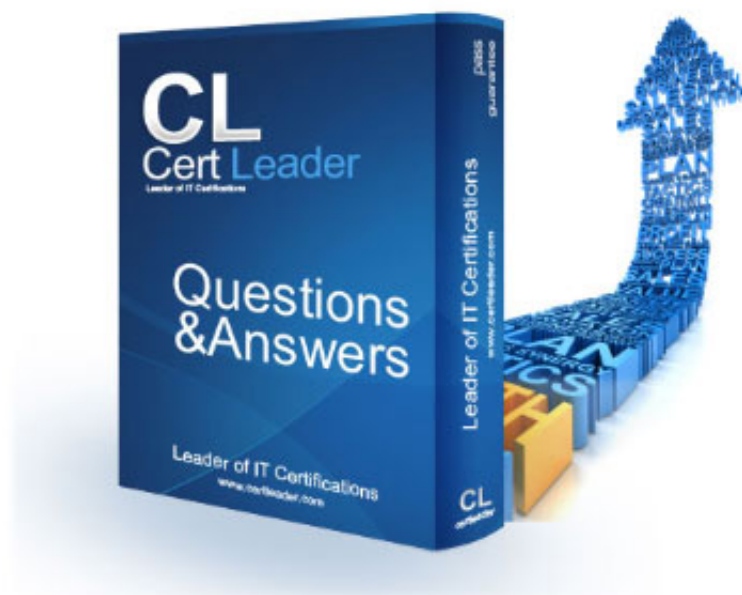


CKA Dumps

Certified Kubernetes Administrator (CKA) Program

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



NEW QUESTION 1

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 lookup the DNS records of the service & pod and write the output to `/opt/KUNW00601/service.dns` and `/opt/KUNW00601/pod.dns` respectively.

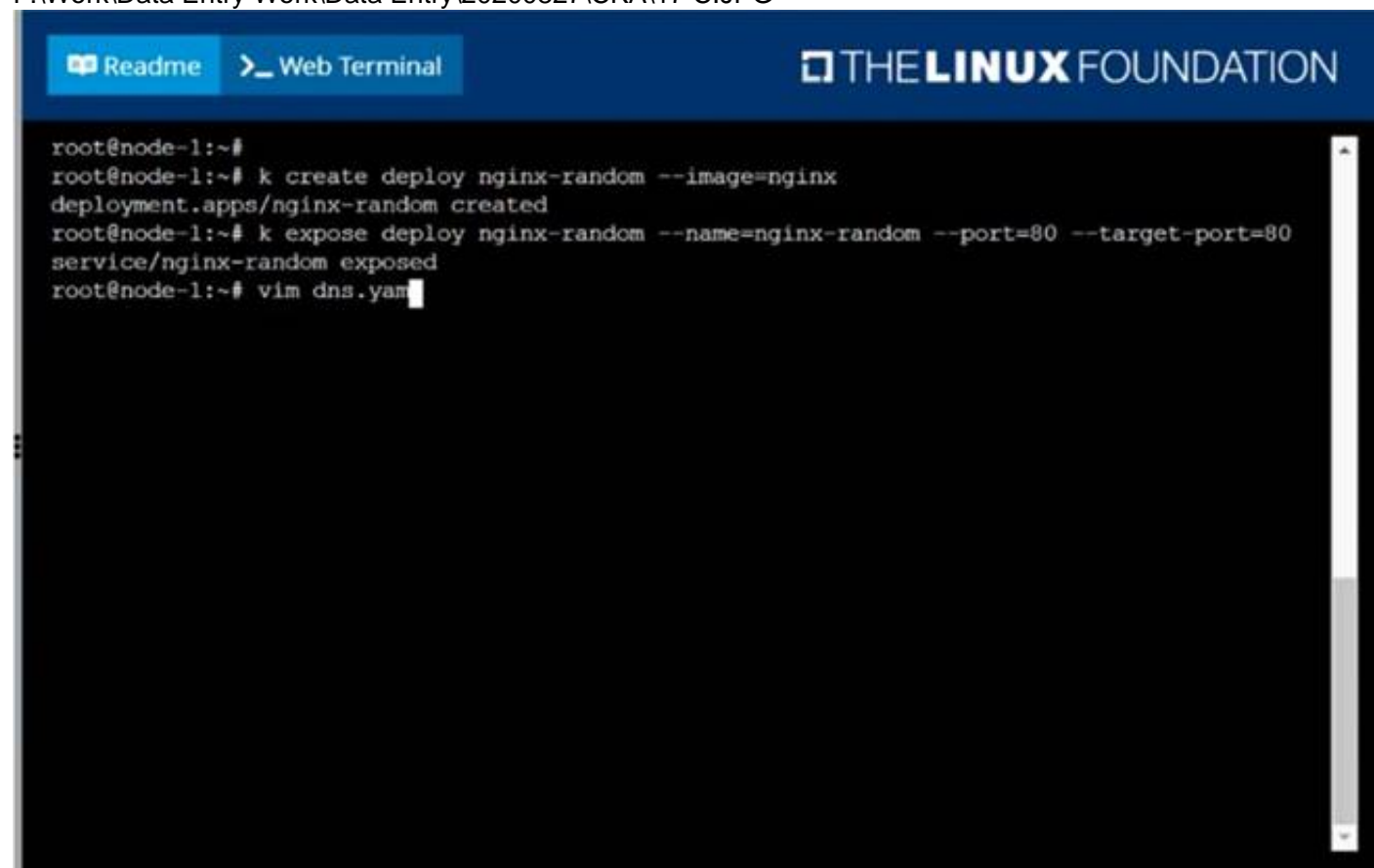
- A. Mastered
- B. Not Mastered

Answer: A

Explanation:

Solution:

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



```
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 D.JPG



```
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 E.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

```

NEW QUESTION 2

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 3

Create a deployment spec file that will:

- > Launch 7 replicas of the nginx image with the label app_runtime_stage=dev
- > deployment name: kua100201

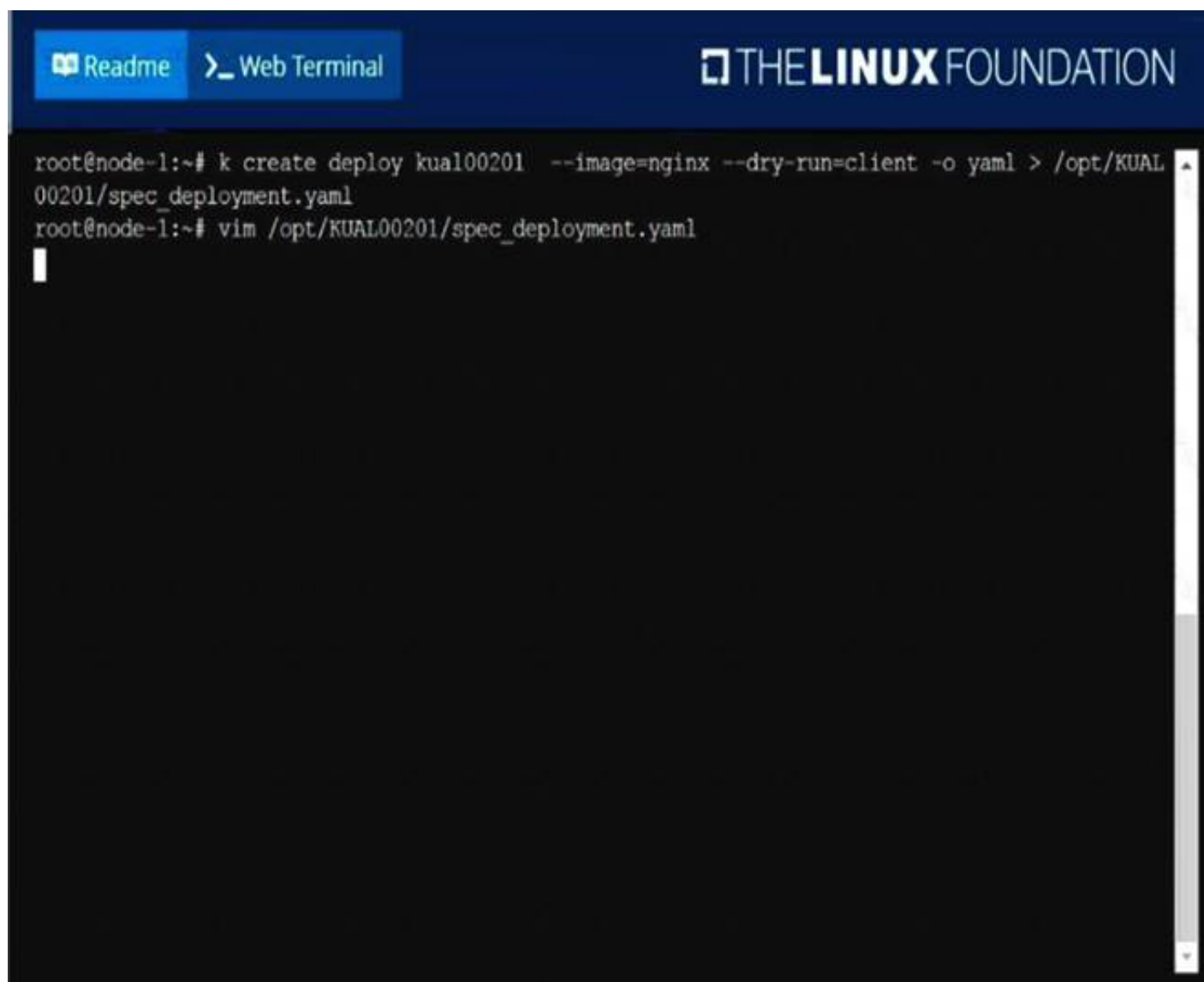
Save a copy of this spec file to /opt/KUAL00201/spec_deployment.yaml (or /opt/KUAL00201/spec_deployment.json).
When you are done, clean up (delete) any new Kubernetes API object that you produced during this task.

- A. Mastered
- B. Not Mastered

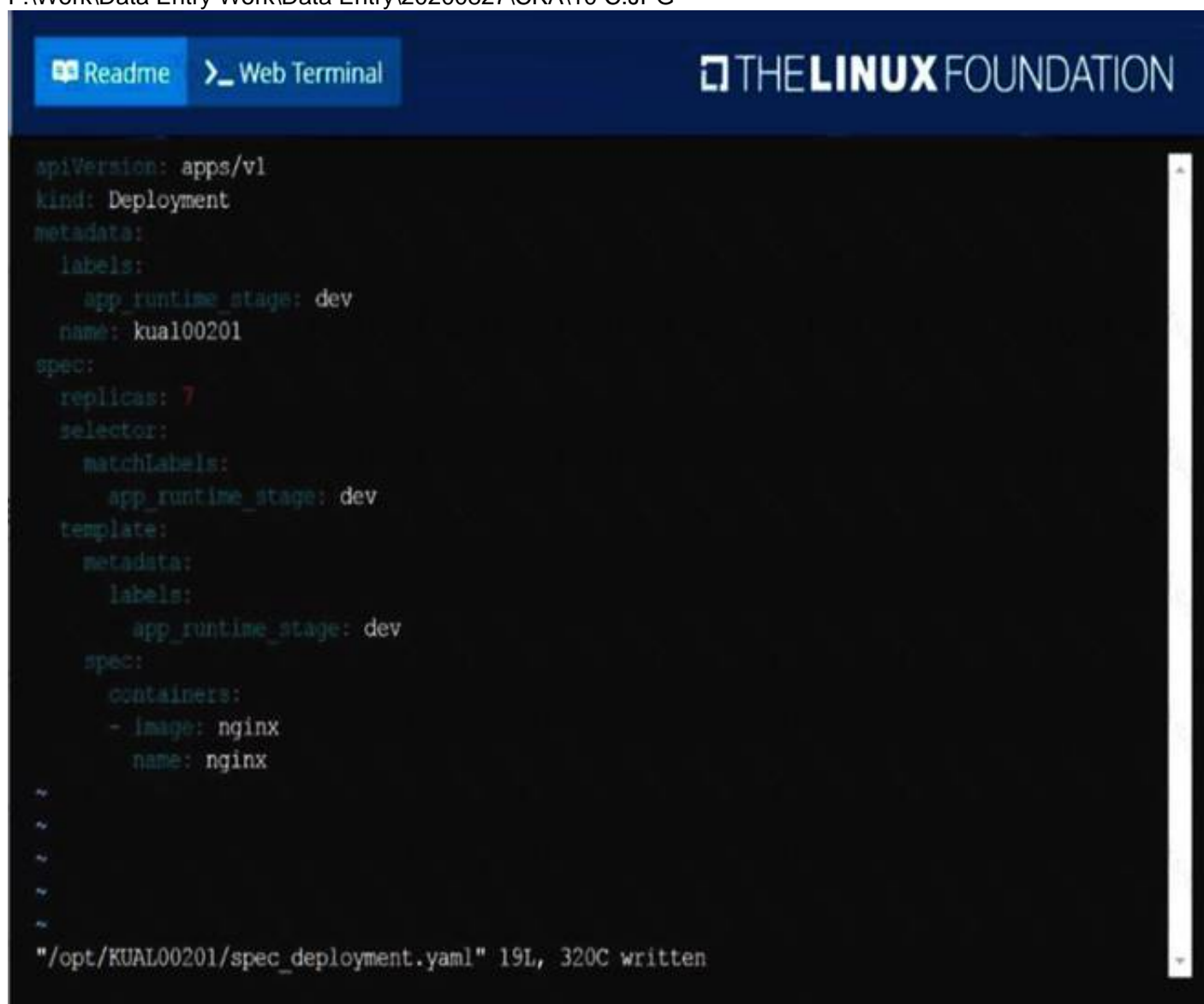
Answer: A

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 4

Create a pod as follows:

- > Name:mongo
- > Using Image:mongo
- > In anew Kubernetes namespacenamed:my-website

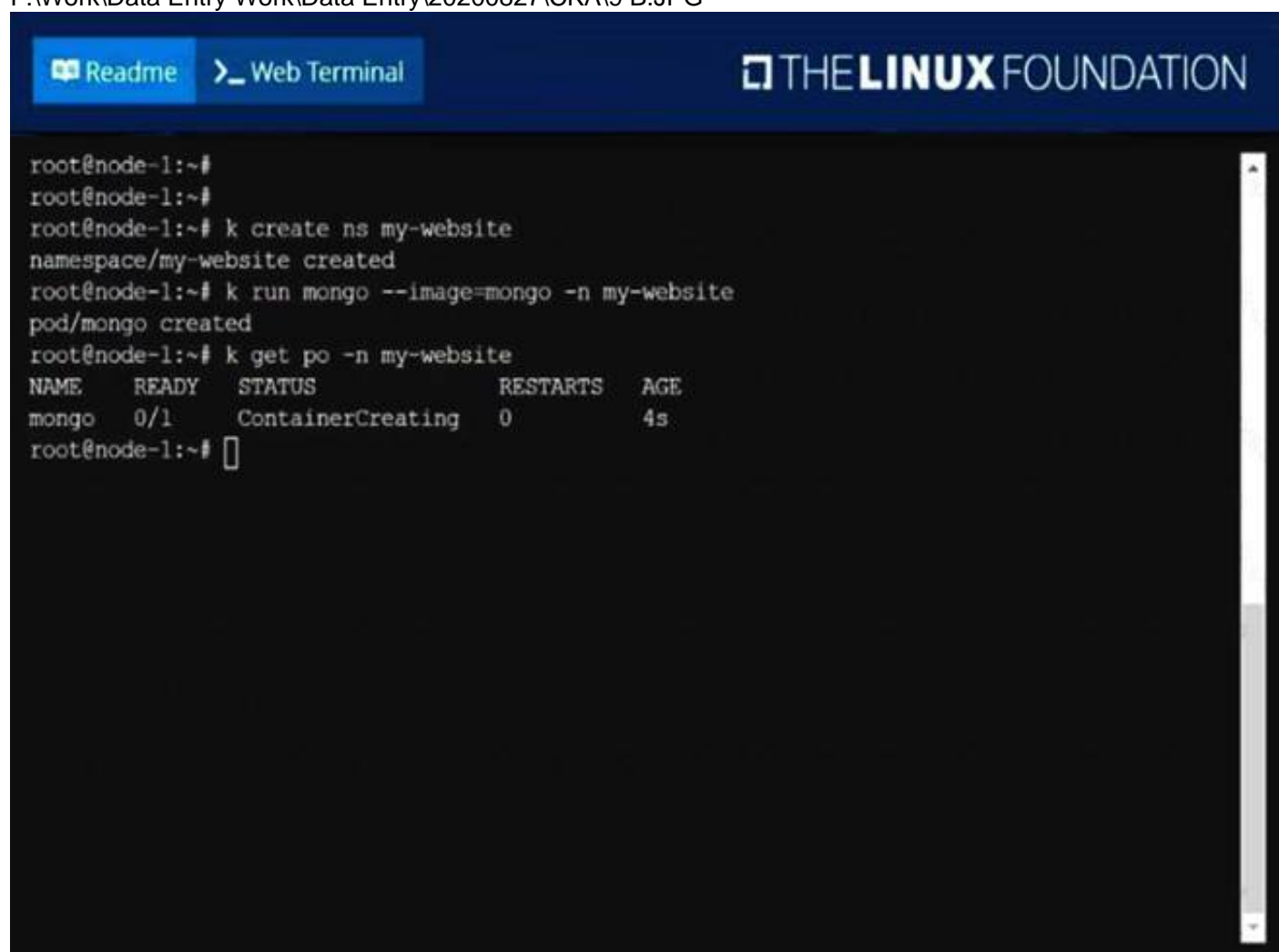
- A. Mastered
- B. Not Mastered

Answer: A

Explanation:

solution

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



The screenshot shows a terminal window with a dark background and light-colored text. At the top, there is a header bar with a blue background containing the text 'Readme' and 'Web Terminal' with a magnifying glass icon. To the right of this bar is the 'THE LINUX FOUNDATION' logo. The terminal content shows a series of commands and their outputs:

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

NEW QUESTION 5

List ??nginx-dev?? and ??nginx-prod?? pod and delete those pods

- A. Mastered
- B. Not Mastered

Answer: A

Explanation:

kubect1 get pods -o wide

kubectl delete po ??nginx-dev??kubectl delete po ??nginx-prod??

NEW QUESTION 6

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

- A. Mastered
- B. Not Mastered

Answer: A

Explanation:

kubectlrn nginx --image=nginx --restart=Never --port=80

NEW QUESTION 7

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 8

Ensure a single instance of podnginxis running on each node of theKubernetes cluster wherenginxalso represents the Image name whichhas to be used. Do not override anytaints currently in place.

UseDaemonSetto complete thistask and useds-kusc00201asDaemonSet name.

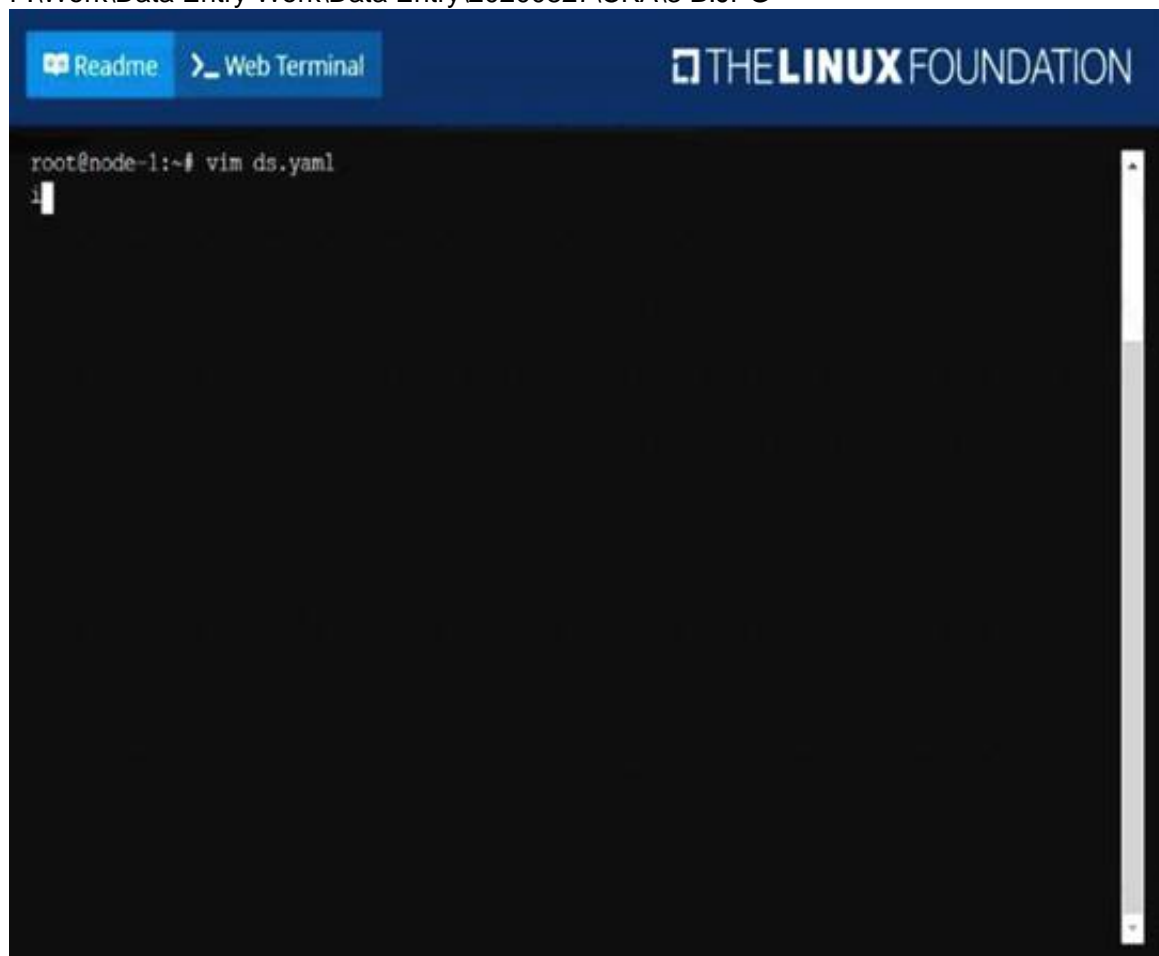
- A. Mastered
- B. Not Mastered

Answer: A

Explanation:

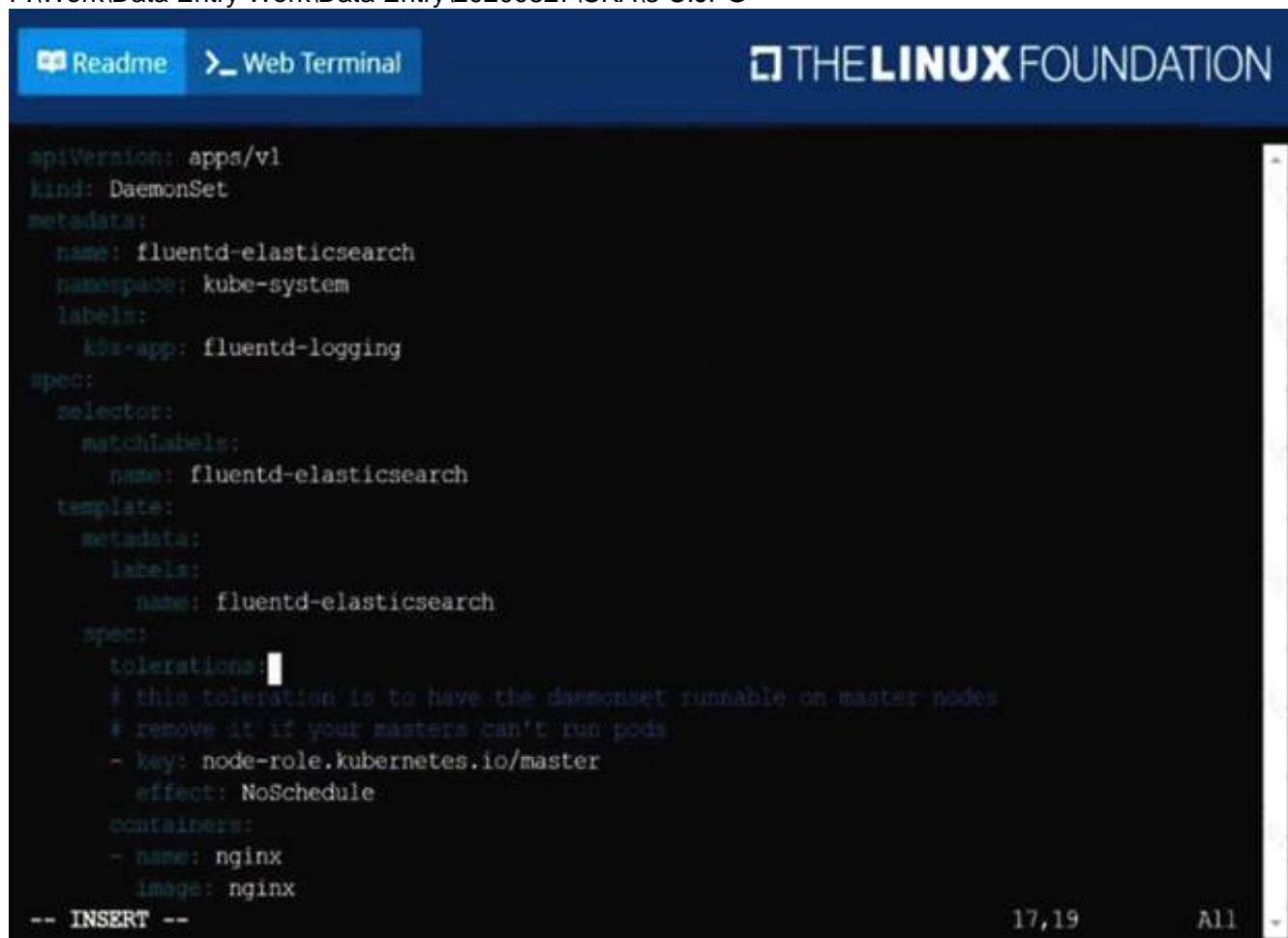
solution

F:\Work\Data Entry Work\Data Entry\20200827\CKA\3 B.JPG



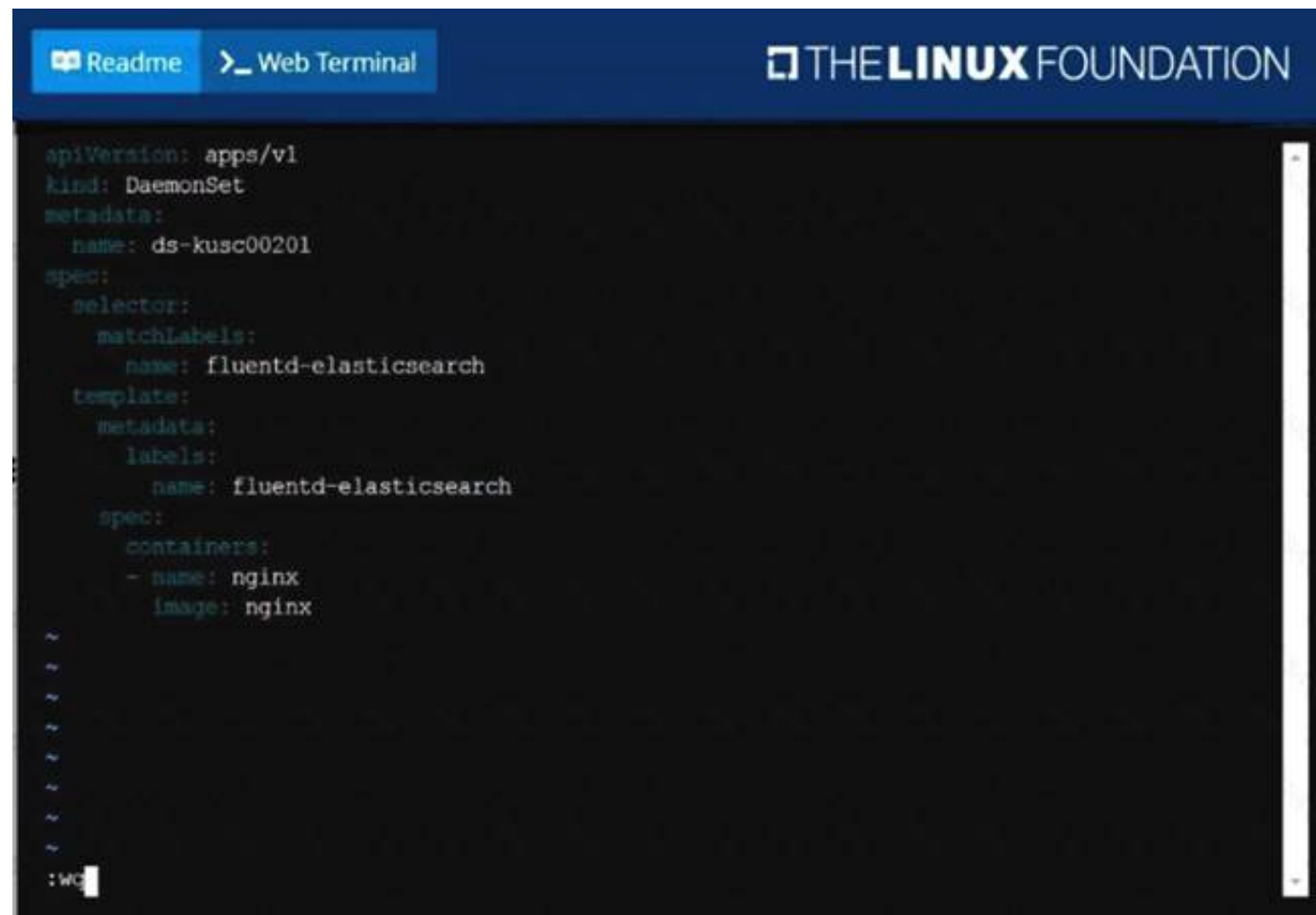
```
root@node-1:~# vim ds.yaml
:
```

F:\Work\Data Entry Work\Data Entry\20200827\CKA\3 C.JPG

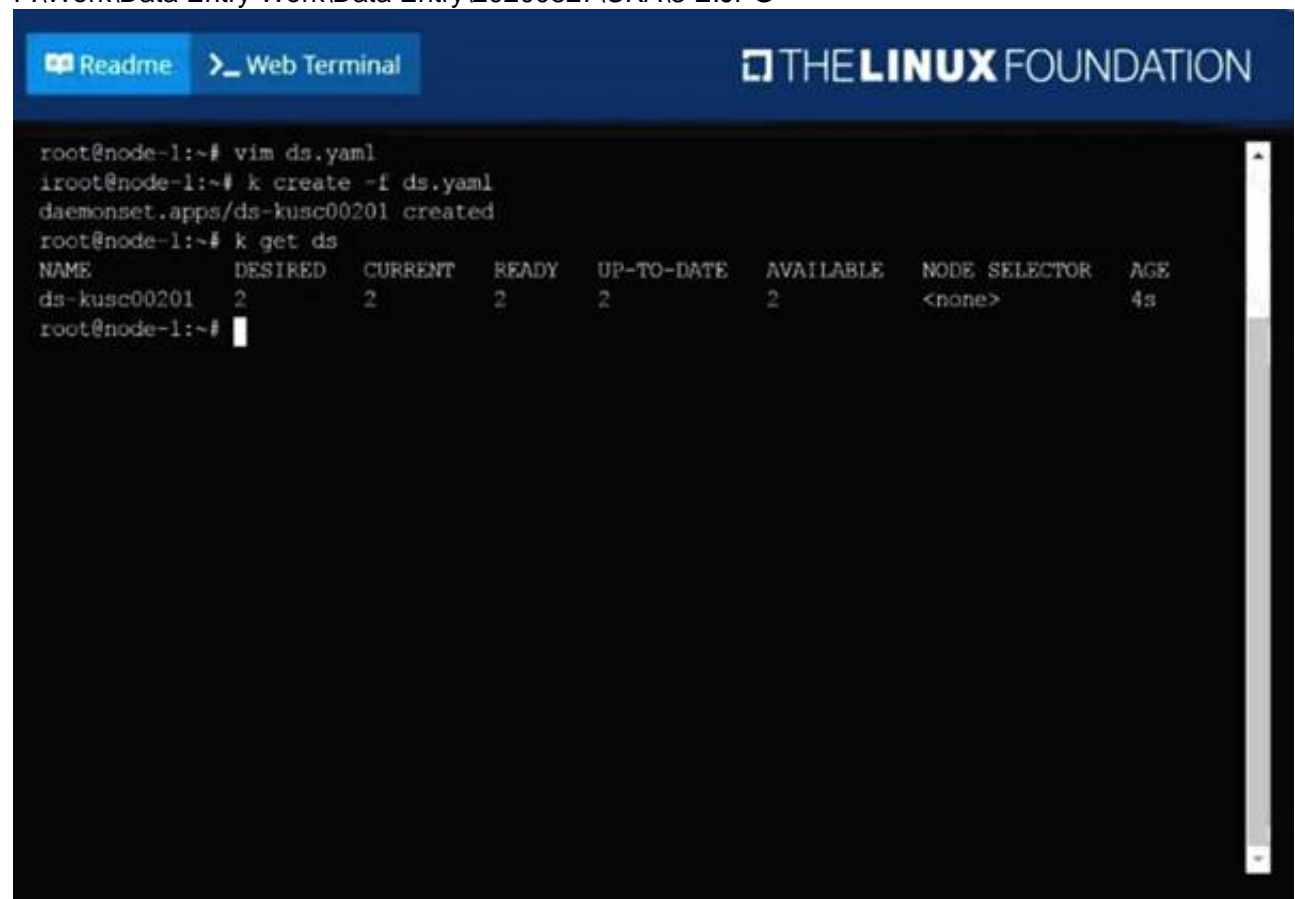


```
apiVersion: apps/v1
kind: DaemonSet
metadata:
  name: fluentd-elasticsearch
  namespace: kube-system
  labels:
    k8s-app: fluentd-logging
spec:
  selector:
    matchLabels:
      name: fluentd-elasticsearch
  template:
    metadata:
      labels:
        name: fluentd-elasticsearch
    spec:
      tolerations:
        # this toleration is to have the daemonset runnable on master nodes
        # remove it if your masters can't run pods
        - key: node-role.kubernetes.io/master
          effect: NoSchedule
      containers:
        - name: nginx
          image: nginx
-- INSERT --
```

F:\Work\Data Entry Work\Data Entry\20200827\CKA\3 D.JPG



F:\Work\Data Entry Work\Data Entry\20200827\CKA\3 E.JPG



NEW QUESTION 9

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

You can `ssh` to the failed node using:

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

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

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

- A. Mastered
- B. Not Mastered

Answer: A

Explanation:

solution

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

```
Readme Web Terminal THE LINUX FOUNDATION

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

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

```
Readme Web Terminal THE LINUX FOUNDATION

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

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

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

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

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

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

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

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



The screenshot shows a web terminal interface with a dark background. At the top, there are two tabs: 'Readme' and 'Web Terminal'. The 'Web Terminal' tab is active, displaying a series of terminal commands and their outputs. The commands include checking for updates, upgrading the system, restarting and enabling kubelet, and listing the nodes in the cluster. The output shows that the cluster is healthy with three nodes (one master and two workers) all in a 'Ready' state.

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

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

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

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

NEW QUESTION 10

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 10

Check the image version in pod without the describe command

- A. Mastered
- B. Not Mastered

Answer: A

Explanation:

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

NEW QUESTION 11

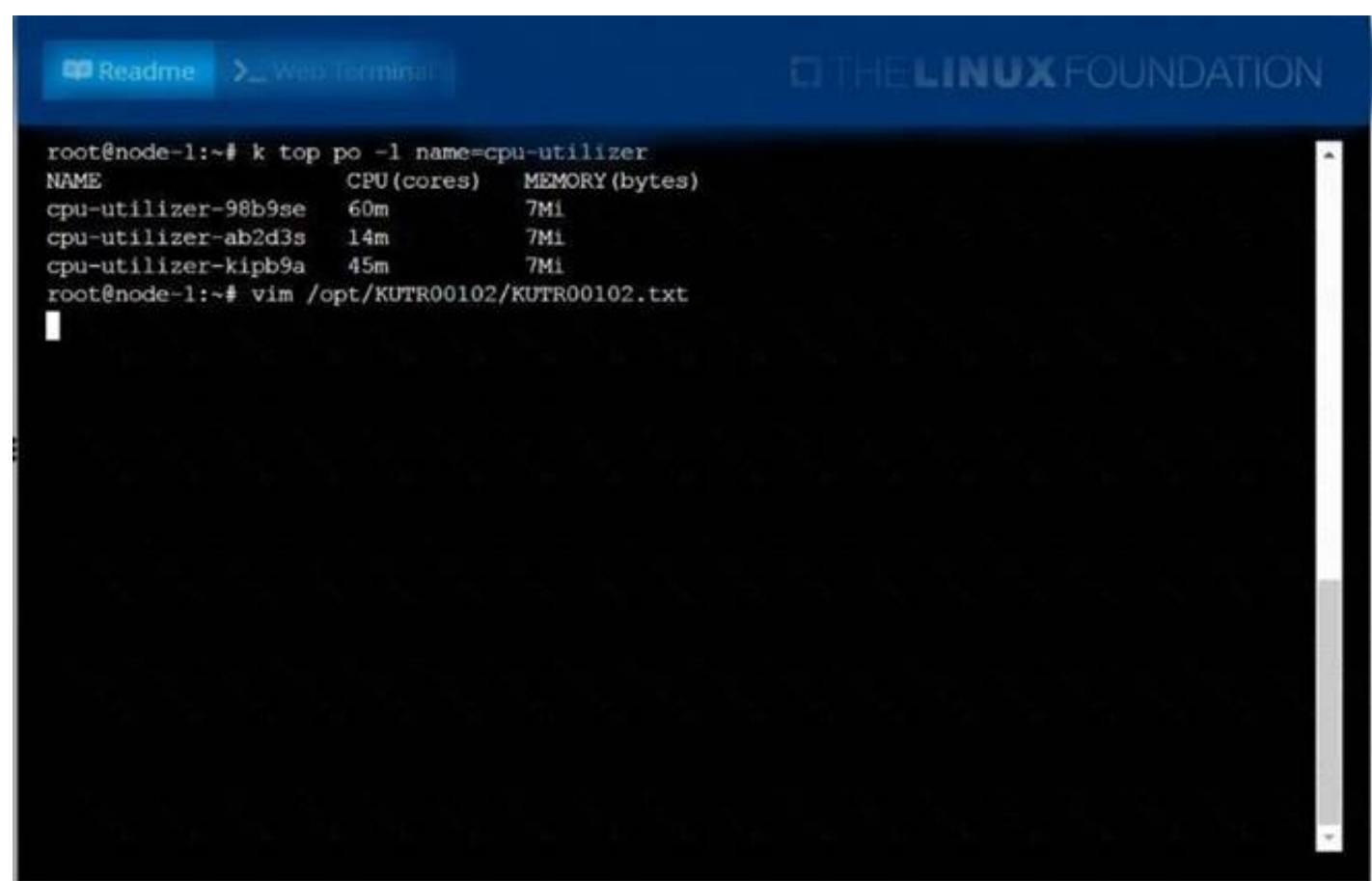
From the pod labelname=cpu-utilizer, find podsrunning high CPU workloads and write the name of the pod consumingmost CPU to thefile/opt/KUTR00102/KUTR00102.txt(which already exists).

- A. Mastered
- B. Not Mastered

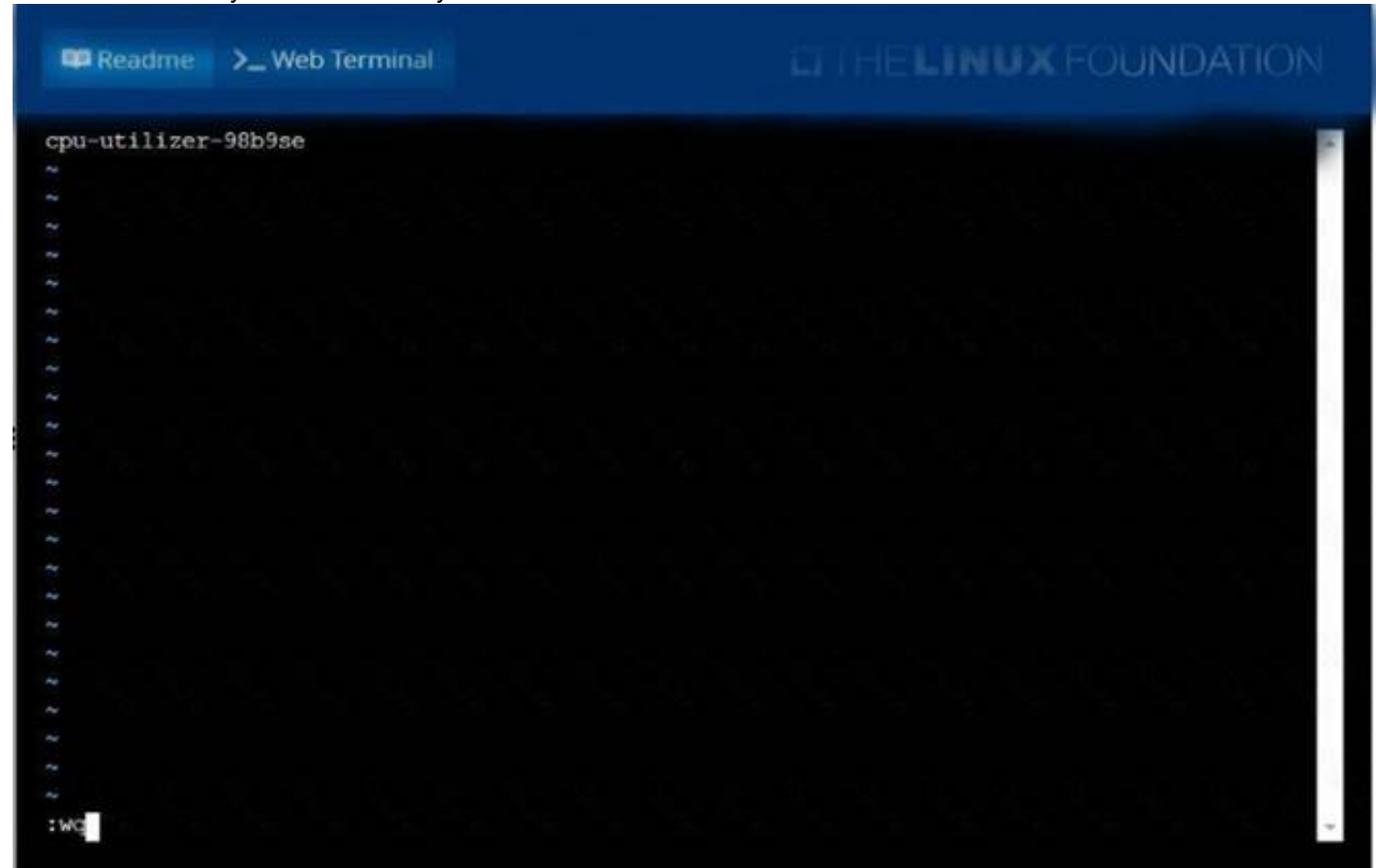
Answer: A

Explanation:

solution
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 16

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

- A. Mastered
- B. Not Mastered

Answer: A

Explanation:

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

```

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:~#

```

NEW QUESTION 19

For this item, you will have to ssh to the node `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=alloption`.

- > 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 23

Get IP address of the pod `?C ??nginx-dev??`

- A. Mastered
- B. Not Mastered

Answer: A

Explanation:

Kubectl `get po -o wide` Using JsonPath

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

NEW QUESTION 26

List all the pods showing name and namespace with a json path expression

- A. Mastered
- B. Not Mastered

Answer: A

Explanation:

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

NEW QUESTION 30

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`:

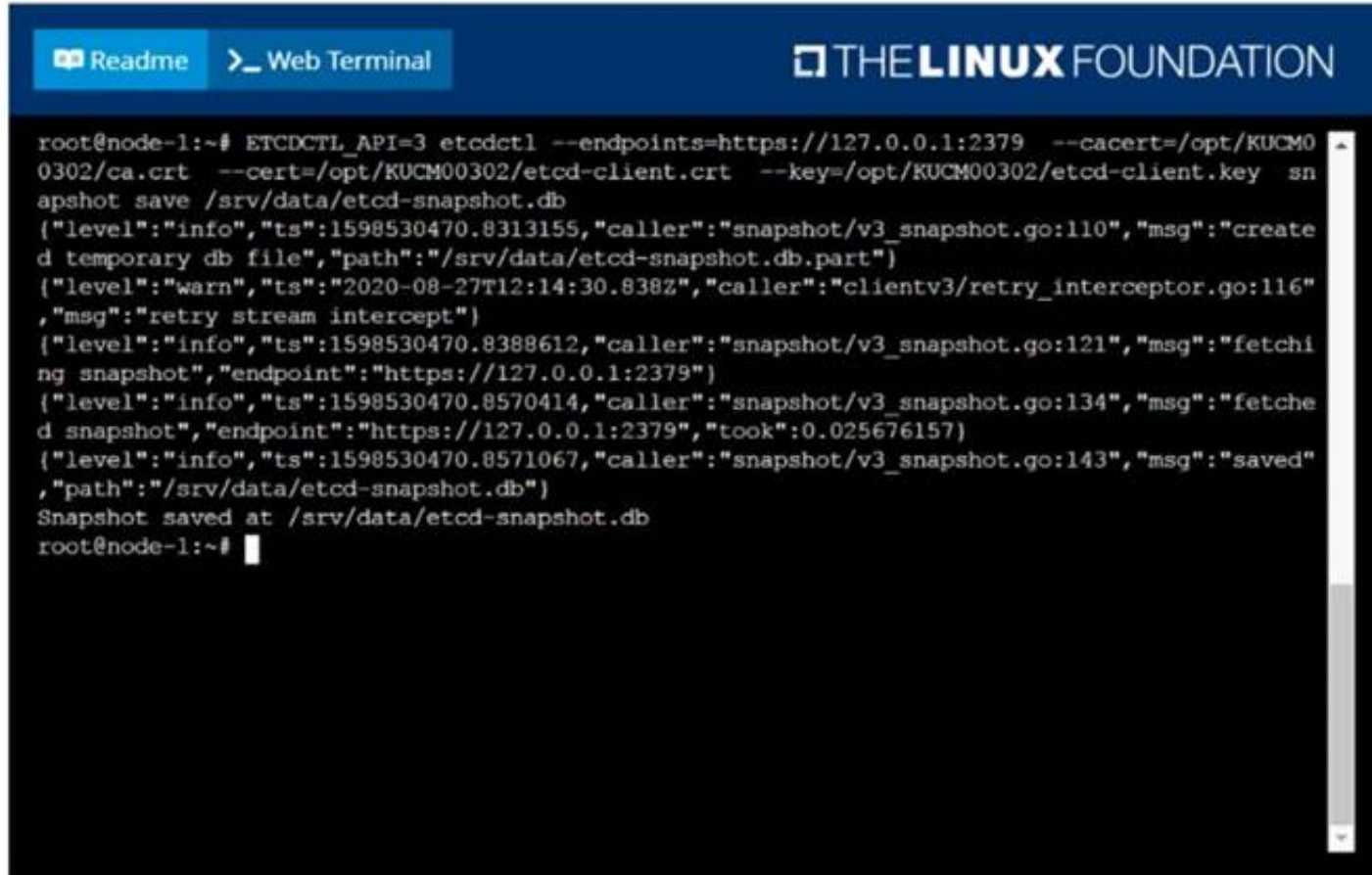
- CA certificate:/opt/KUCM00302/ca.crt
- Client certificate:/opt/KUCM00302/etcd-client.crt
- Client key:Topt/KUCM00302/etcd-client.key

- A. Mastered
B. Not Mastered

Answer: A

Explanation:

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



NEW QUESTION 31

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 33

Create a pod that having 3 containers in it? (Multi-Container)

- A. Mastered
B. Not Mastered

Answer: A

Explanation:

image=nginx, image=redis, image=consul Name nginx container as ??nginx-container?? Name redis container as ??redis-container?? Name consul container as ??consul-container??

Create a pod manifest file for a container and append container section for rest of the images

kubectl run multi-container --generator=run-pod/v1 --image=nginx -- dry-run -o yaml > multi-container.yaml

then

vim multi-container.yaml apiVersion: v1

kind: Pod metadata: labels:

run: multi-container name: multi-container spec:

containers:

- image: nginx

name: nginx-container

- image: redis

name: redis-container

- image: consul

name: consul-container

restartPolicy: Always

NEW QUESTION 36

.....

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>