Certified Kubernetes Administrator (CKA) Program Exam

Linux Foundation CKA Dumps Available Here at:

https://www.certification-questions.com/linux-foundation-exam/cka-dumps.html

Enrolling now you will get access to 53 questions in a unique set of CKA dumps

Question 1

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

Options:

A. Option is correct.
See the solution below.

Explanation:

kubectl run --generator=run-pod/v1 --image=nginx -- labels=env=prod nginx-prod --dry-run -o yaml >
nginx-prodpod.yaml Now, edit nginx-prod-pod.yaml file and remove entries like "creationTimestamp:
null"
"dnsPolicy: ClusterFirst"
vim nginx-prod-pod.yaml
apiVersion: v1
kind: Pod
metadata:
labels:
env: prod
name: nginx-prod
spec:
containers:
- image: nginx
name: nginx-prod

restartPolicy: Always
kubectl create -f nginx-prod-pod.yaml
kubectl run --generator=run-pod/v1 --image=nginx --
labels=env=dev nginx-dev --dry-run -o yaml > nginx-dev-pod.yaml

apiVersion: v1
kind: Pod
metadata:
labels:
env: dev
name: nginx-dev
spec:
containers:
- image: nginx
name: nginx-dev
restartPolicy: Always
kubectl create -f nginx-prod-dev.yaml
Verify :
kubectl get po --show-labels
kubectl get po -l env=prod
kubectl get po -l env=dev

Answer: A

Explanation:
rrect.
See the solution below.

Explanation:

kubectl run --generator=run-pod/v1 --image=nginx -- labels=env=prod nginx-prod --dry-run -o yaml > nginx-prodpod.yaml Now, edit nginx-prod-pod.yaml file and remove entries like "creationTimestamp: null"
"dnsPolicy: ClusterFirst"
vim nginx-prod-pod.yaml
apiVersion: v1
kind: Pod
metadata:

labels:
env: prod
name: nginx-prod
spec:
containers:
- image: nginx
name: nginx-prod
restartPolicy: Always
kubectl create -f nginx-prod-pod.yaml
kubectl run --generator=run-pod/v1 --image=nginx --
labels=env=dev nginx-dev --dry-run -o yaml > nginx-dev-pod.yaml

apiVersion: v1
kind: Pod
metadata:
labels:
env: dev
name: nginx-dev
spec:
containers:
- image: nginx
name: nginx-dev
restartPolicy: Always
kubectl create -f nginx-prod-dev.yaml
Verify :
kubectl get po --show-labels
kubectl get po -l env=prod
kubectl get po -l env=dev

Question 2

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

Options:

A. Option is correct.
See the solution below.

Explanation:

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

Answer: A

Explanation:
rrect.
See the solution below.

Explanation:

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

Question 3

Create a pod with environment variables as var1=value1.Check the environment variable in pod

Options:

A. Option is correct.
See the solution below.

Explanation:

kubectl run nginx --image=nginx --restart=Never --env=var1=value1
then
kubectl exec -it nginx -- env
or
kubectl exec -it nginx -- sh -c 'echo $var1'
or
kubectl describe po nginx | grep value1

Answer: A

Explanation:
rrect.
See the solution below.

Explanation:

kubectl run nginx --image=nginx --restart=Never --env=var1=value1
then
kubectl exec -it nginx -- env
or

kubectl exec -it nginx -- sh -c 'echo $var1'
or
kubectl describe po nginx | grep value1

Question 4

List all the pods sorted by name

Options:

A. Option is correct.
See the solution below.

Explanation:

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

Answer: A

Explanation:
rrect.
See the solution below.

Explanation:

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

Question 5

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

Options:

A. Option is correct.
See the solution below.

Explanation:

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

Answer: A

Explanation:
rrect.
See the solution below.

Explanation:

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

Question 6

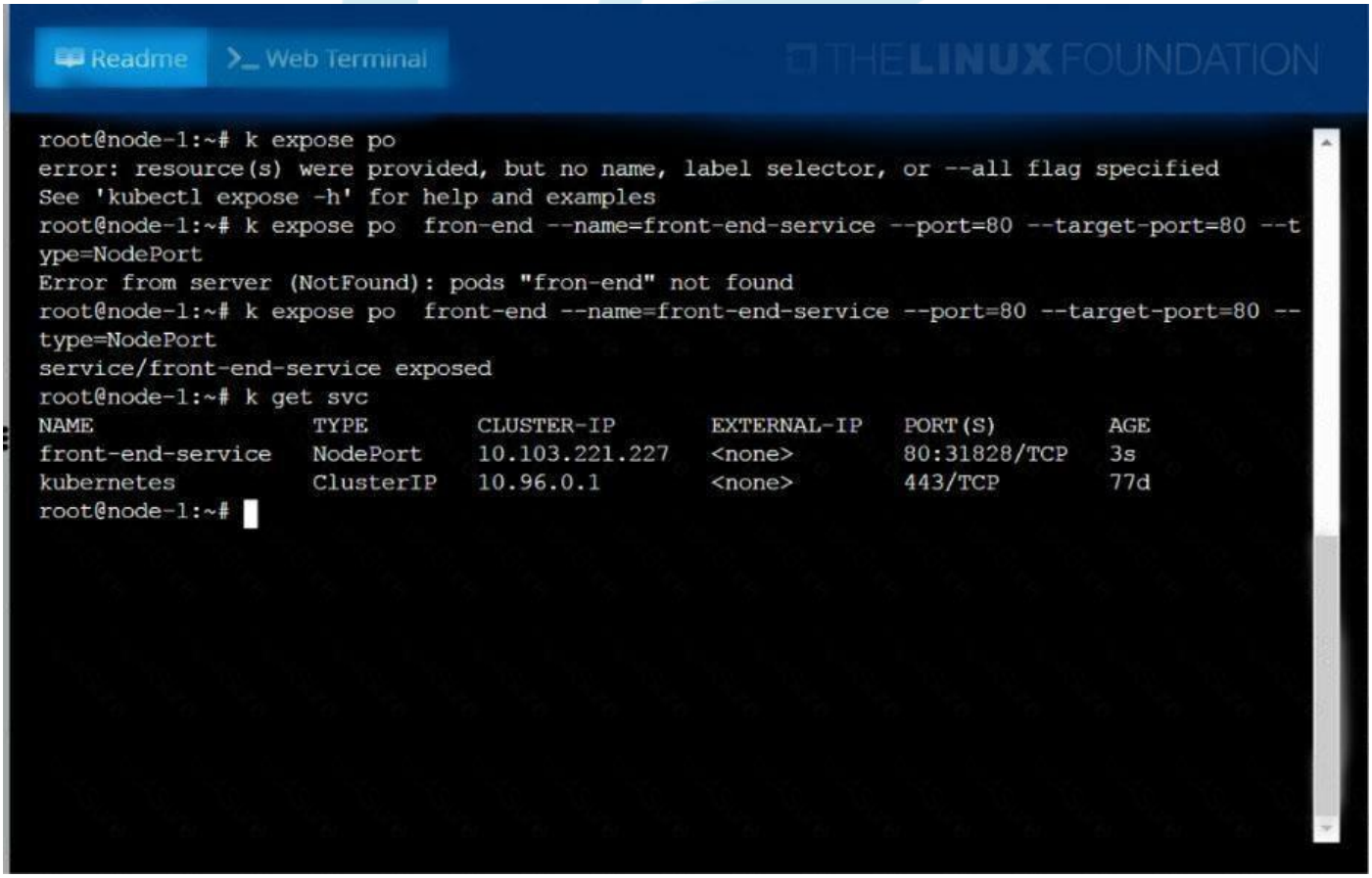
Create and configure the servicefront-end-serviceso it's accessiblethroughNodePortand routes to theexisting pod namedfront-end.

Options:

A. Option is correct.
See the solution below.

Explanation:

solution

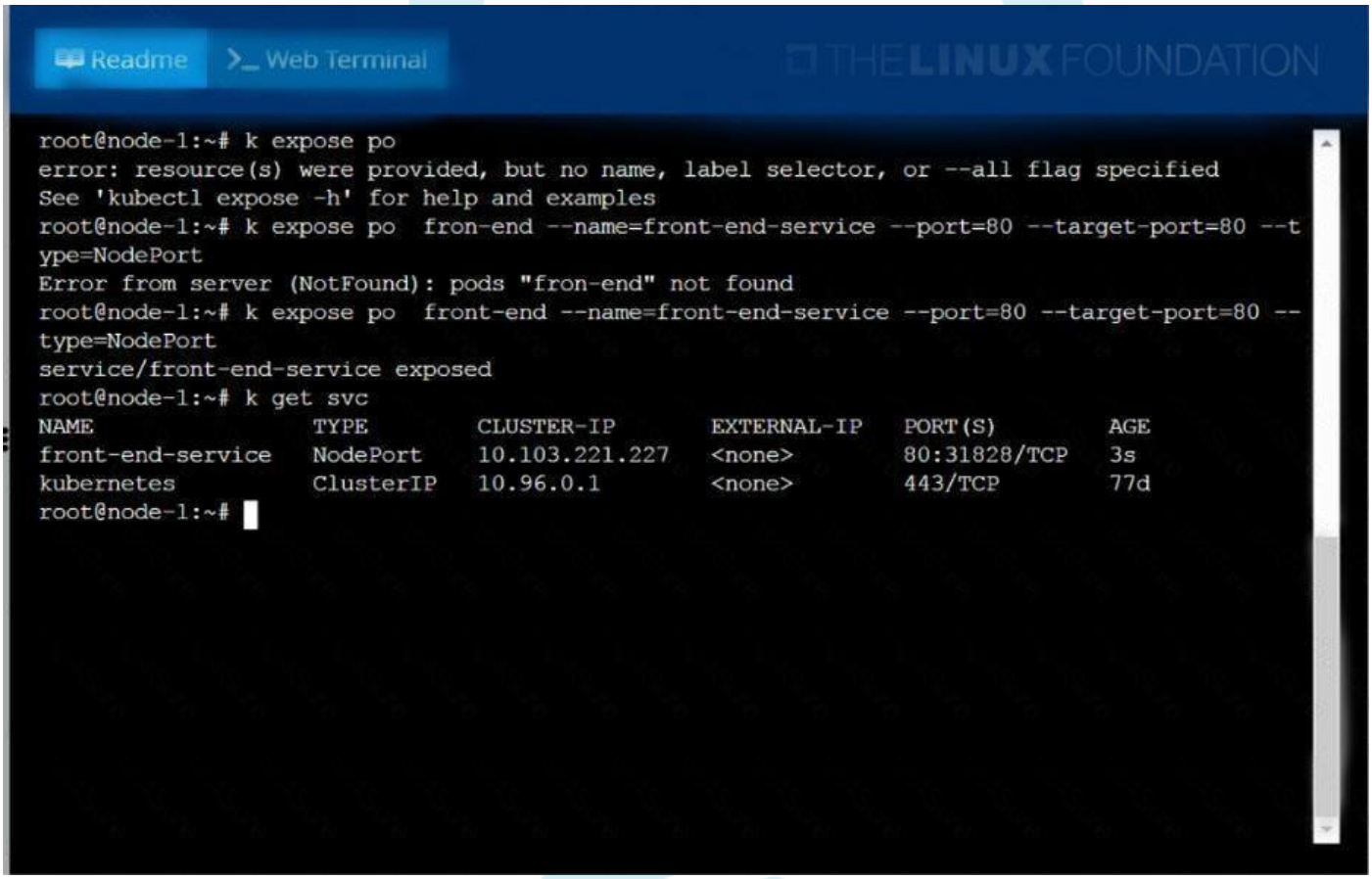


Answer: A

Explanation:
rrect.
See the solution below.

Explanation:

solution



Question 7

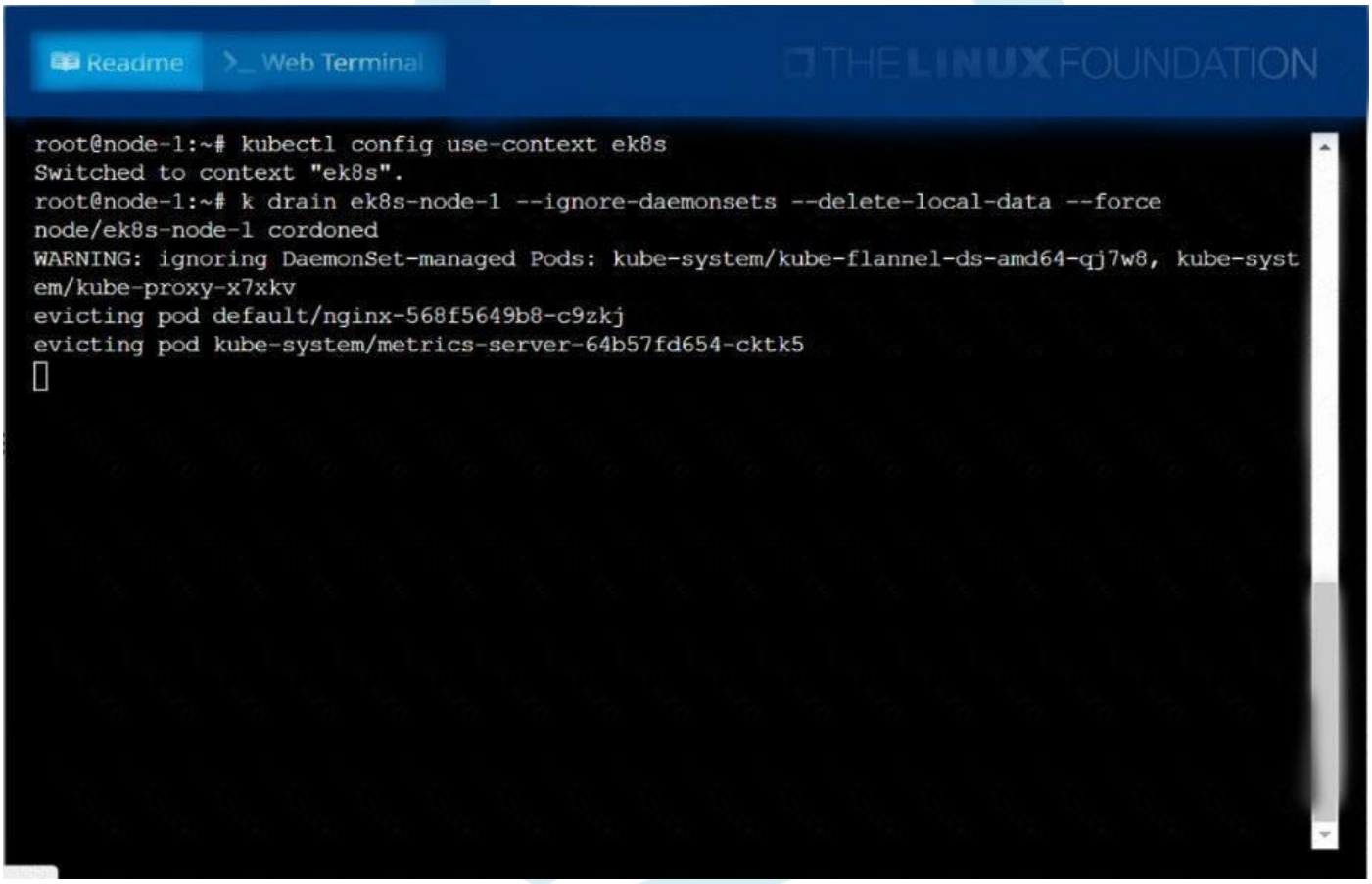
Set the node namedek8s-node-1asunavailable and reschedule all thepods running on it.

Options:

A. Option is correct.
See the solution below.

Explanation:

solution

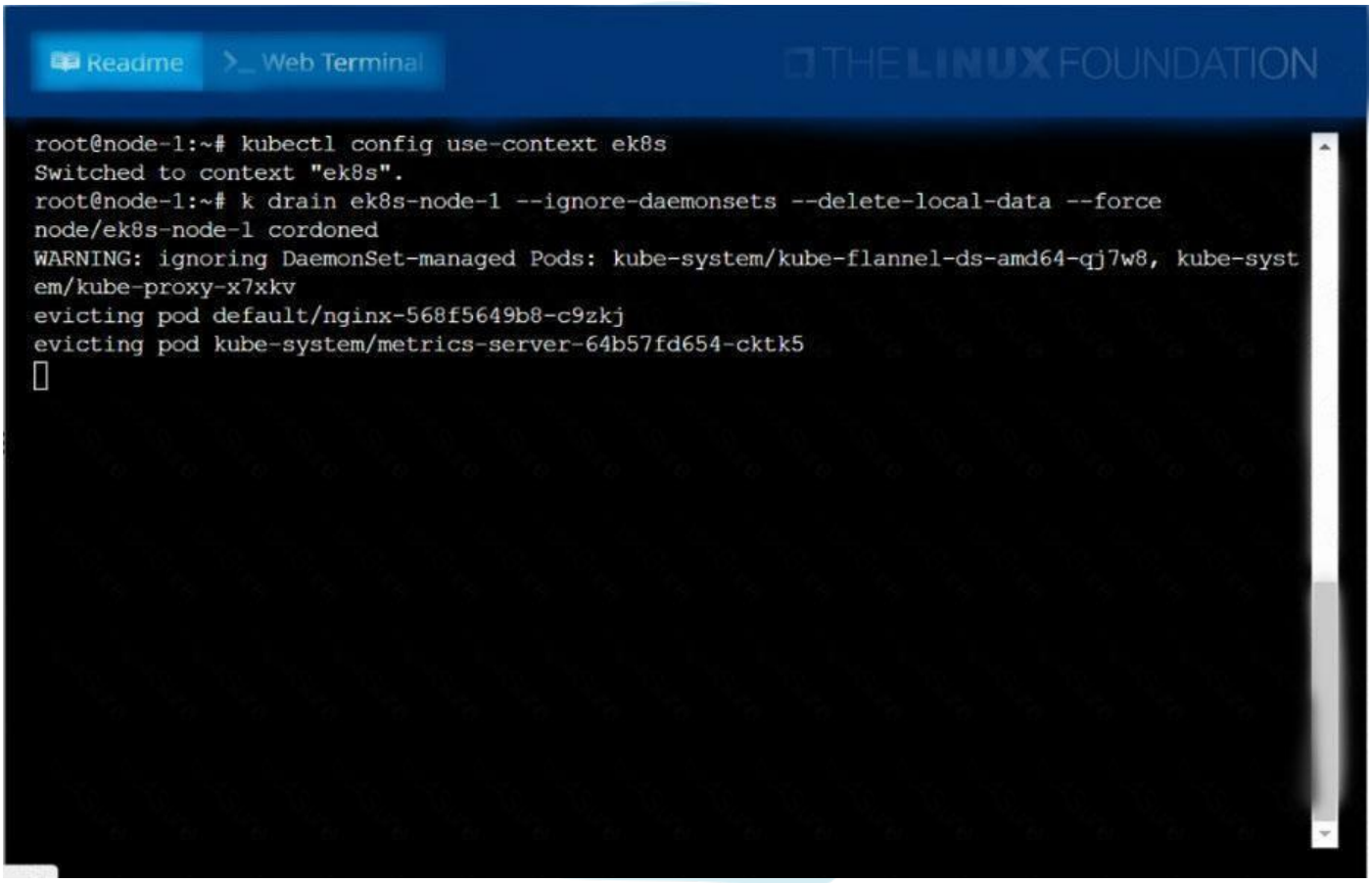


Answer: A

Explanation:
rrect.
See the solution below.

Explanation:

solution



Question 8

Create a busybox pod that runs the command "env" and save the output to "envpod" file

Options:

A. Option is correct.
See the solution below.

Explanation:

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

Answer: A

Explanation:
rrect.
See the solution below.

Explanation:

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

Question 9

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

Options:

A. Option is correct.
See the solution below.

Explanation:

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

Answer: A

Explanation:
rrect.
See the solution below.

Explanation:

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

Question 10

Create an nginx pod with container Port 80 and it should only receive traffic only it checks the endpoint / on port 80 and verify and delete the pod.

Options:

A. kubectl run nginx --image=nginx --restart=Never --port=80 --

dry-run -o yaml > nginx-pod.yaml
// add the readinessProbe section and create
vim nginx-pod.yaml
run: nginx
name: nginx
spec:
containers:
- image: nginx
name: nginx
ports:
- containerPort: 60
readinessProbe:

httpGet:
path: /
port: 60
restartPolicy: Never
kubectl apply -f nginx-pod.yaml
// verify
kubectl describe pod nginx | grep -i readiness
kubectl delete po nginx

B. kubectl run nginx --image=nginx --restart=Never --port=80 --

dry-run -o yaml > nginx-pod.yaml
// add the readinessProbe section and create
vim nginx-pod.yaml
apiVersion: v1
kind: Pod
metadata:
labels:
run: nginx
name: nginx
spec:

containers:
- image: nginx
name: nginx
ports:
- containerPort: 80
readinessProbe:
httpGet:
path: /
port: 80
restartPolicy: Never
kubectl apply -f nginx-pod.yaml
// verify
kubectl describe pod nginx | grep -i readiness
kubectl delete po nginx

Answer: B