Free Questions for CKAD by ebraindumps

Shared by Parrish on 24-05-2024

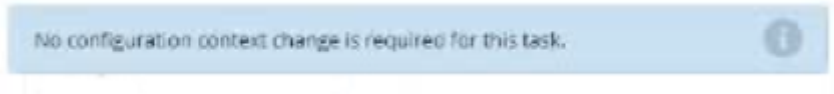
For More Free Questions and Preparation Resources

Check the Links on Last Page

Question 1

Question Type: MultipleChoice

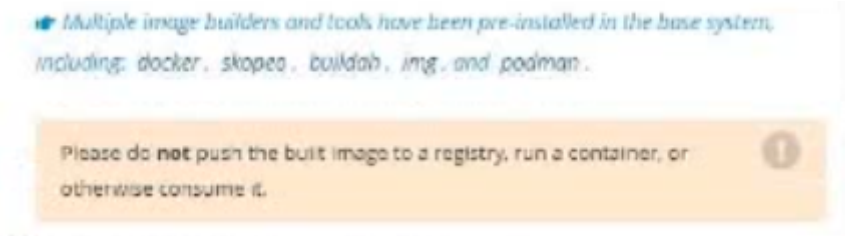
Refer to Exhibit.



Task:

A Dockerfile has been prepared at -/human-stork/build/Dockerfile

1) Using the prepared Dockerfile, build a container image with the name macque and lag 3.0. You may install and use the tool of your choice.

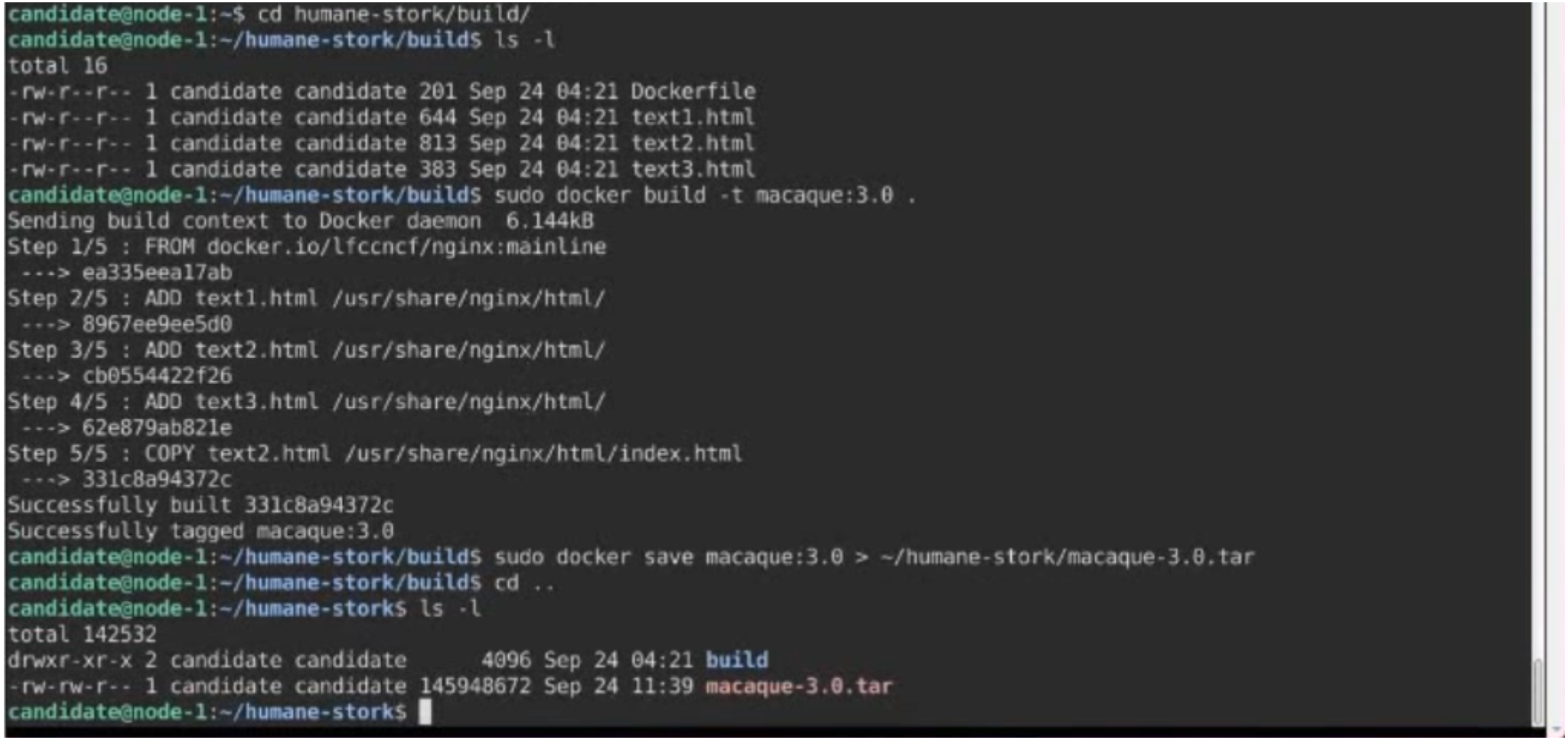


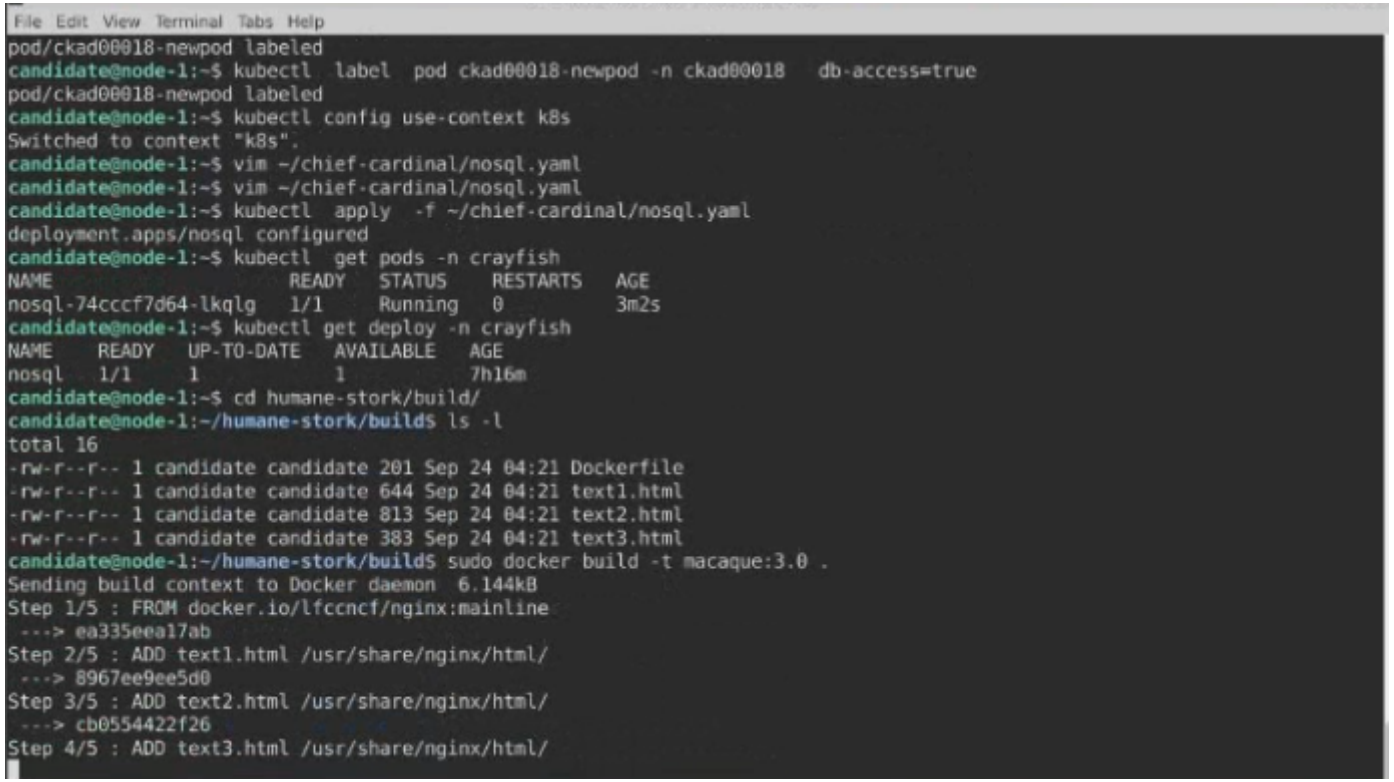
2) Using the tool of your choice export the built container image in OC-format and store it at -/human stork/macque 3.0 tar

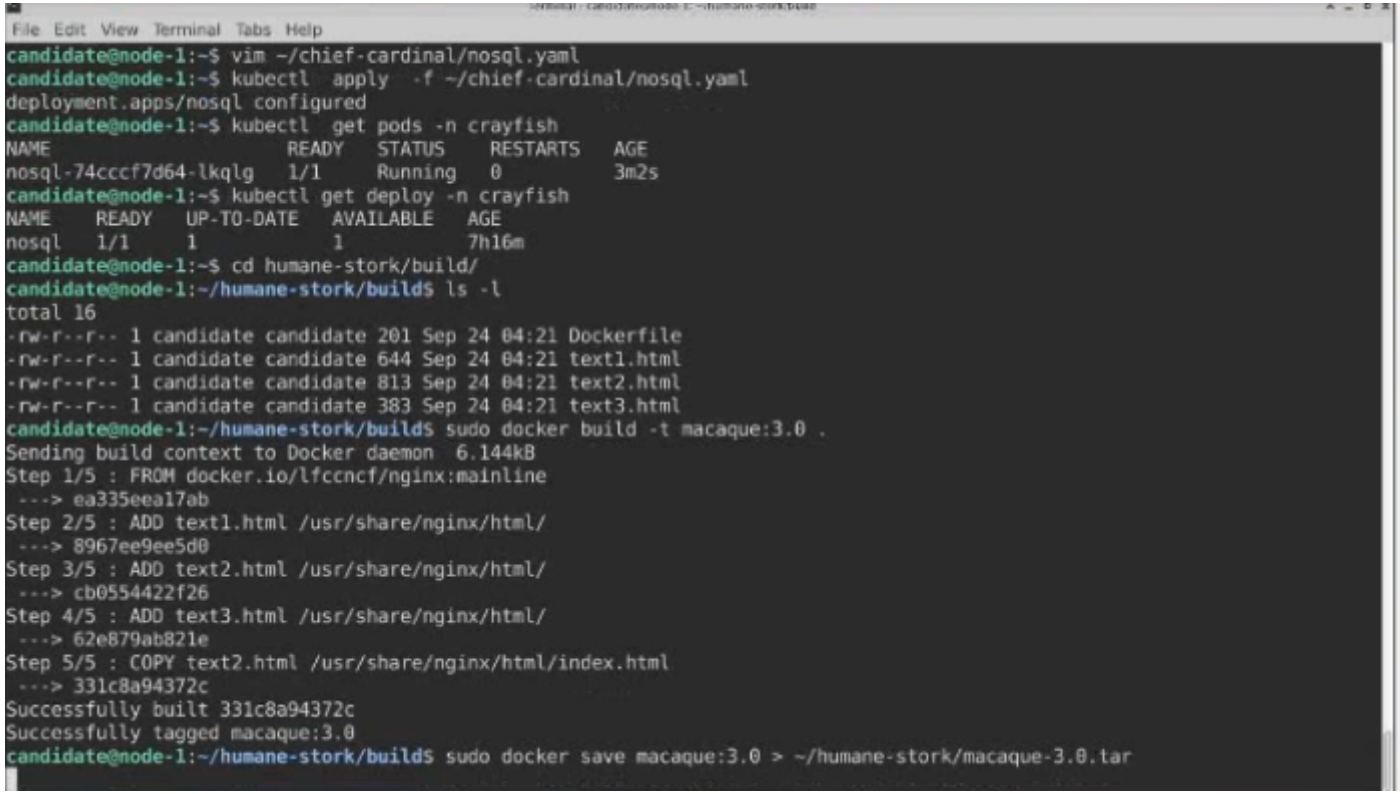
Options:

A- Explanation:

Solution:




File Edit View Terminal Tabs Help
pod/ckad00018-newpod labeled
candidate@node-1:~$ kubectl label pod ckad00018-newpod -n ckad00018   db-access=true
pod/ckad00018-newpod labeled
candidate@node-1:~$ kubectl config use-context k8s
Switched to context "k8s".
candidate@node-1:~$ vim ~/chief-cardinal/nosql.yaml
candidate@node-1:~$ vim ~/chief-cardinal/nosql.yaml
candidate@node-1:~$ kubectl apply -f ~/chief-cardinal/nosql.yaml
deployment.apps/nosql configured
candidate@node-1:~$ kubectl get pods -n crayfish
NAME   READY   STATUS   RESTARTS   AGE
nosql-74cccf7d64-lkqlg   1/1   Running   0   3m2s
candidate@node-1:~$ kubectl get deploy -n crayfish
NAME   READY   UP-TO-DATE   AVAILABLE   AGE
nosql   1/1   1   1   7h16m
candidate@node-1:~$ cd humane-stork/build/
candidate@node-1:~/humane-stork/build$ ls -l
total 16
-rw-r--r-- 1 candidate candidate 201 Sep 24 04:21 Dockerfile
-rw-r--r-- 1 candidate candidate 644 Sep 24 04:21 text1.html
-rw-r--r-- 1 candidate candidate 813 Sep 24 04:21 text2.html
-rw-r--r-- 1 candidate candidate 383 Sep 24 04:21 text3.html
candidate@node-1:~/humane-stork/build$ sudo docker build -t macaque:3.0 .
Sending build context to Docker daemon 6.144kB
Step 1/5 : FROM docker.io/lfccncf/nginx:mainline
 ---> ea335eea17ab
Step 2/5 : ADD text1.html /usr/share/nginx/html/
 ---> 8967ee9ee5d0
Step 3/5 : ADD text2.html /usr/share/nginx/html/
 ---> cb0554422f26
Step 4/5 : ADD text3.html /usr/share/nginx/html/



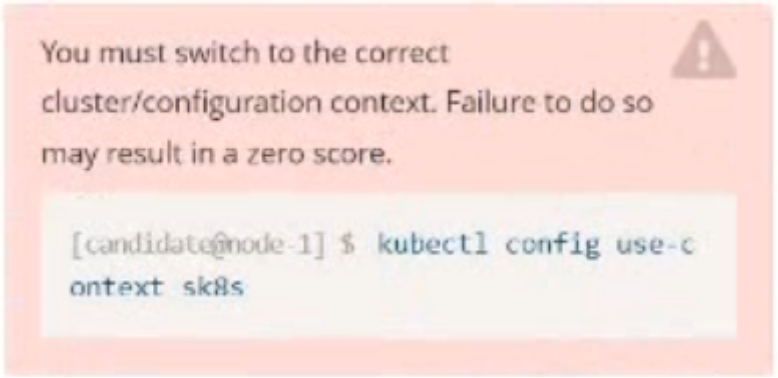
Answer:

A

Question 2

Question Type: **MultipleChoice**

Refer to Exhibit.



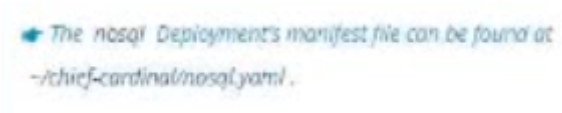
Task:

The pod for the Deployment named nosql in the craytisn namespace fails to start because its container runs out of resources.

Update the nosol Deployment so that the Pod:

1) Request 160M of memory for its Container

2) Limits the memory to half the maximum memory constraint set for the crayfah name space.



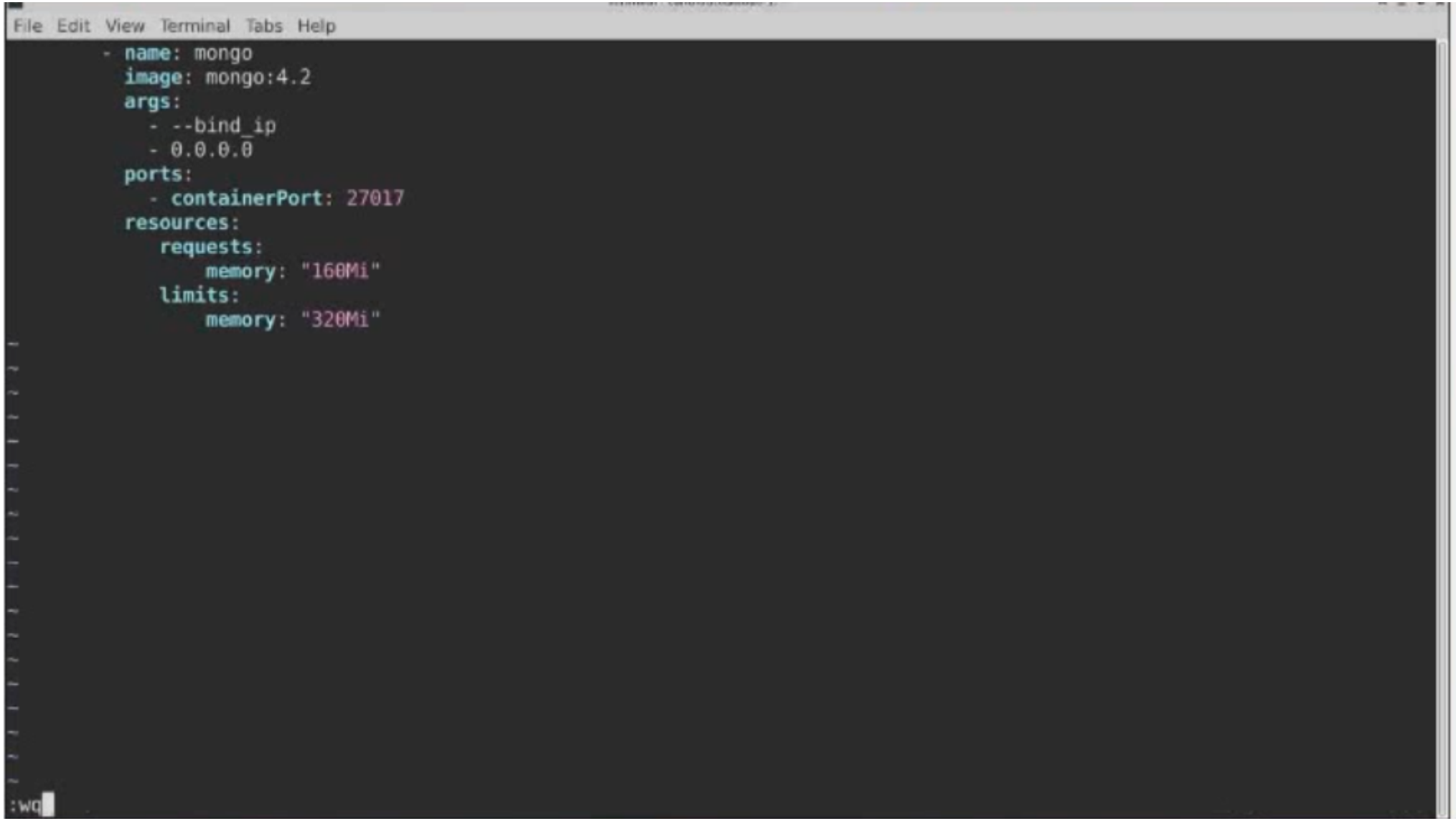
Options:

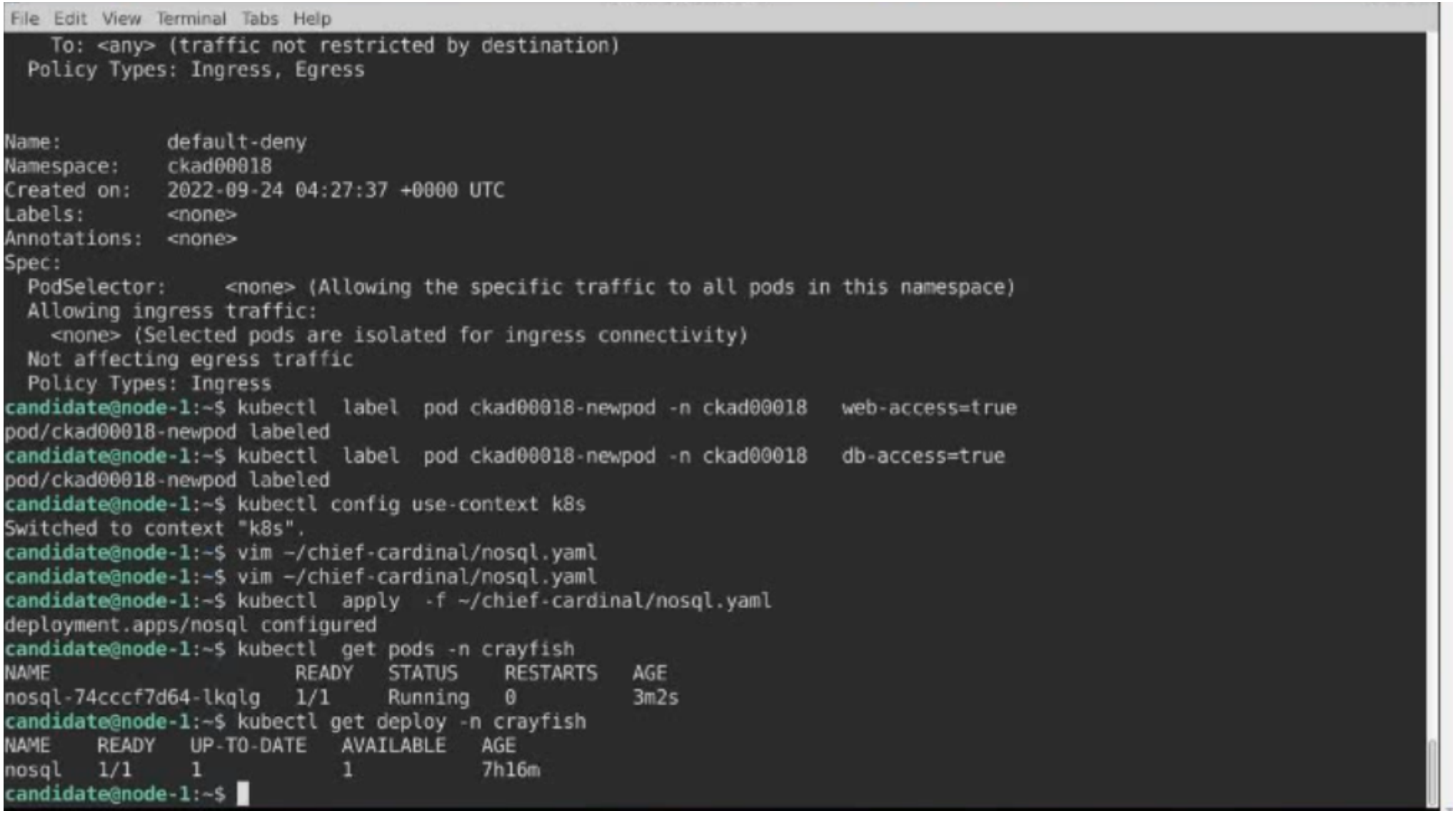
A- Explanation:

Solution:

File Edit View Terminal Tabs Help

```
---
apiVersion: apps/v1
kind: Deployment
metadata:
  name: nosql
  namespace: crayfish
  labels:
    app.kubernetes.io/name: nosql
    app.kubernetes.io/component: backend
spec:
  selector:
    matchLabels:
      app.kubernetes.io/name: nosql
      app.kubernetes.io/component: backend
  replicas: 1
  template:
    metadata:
      labels:
        app.kubernetes.io/name: nosql
        app.kubernetes.io/component: backend
    spec:
      containers:
        - name: mongo
          image: mongo:4.2
          args:
            - --bind_ip
            - 0.0.0.0
          ports:
            - containerPort: 27017
~
~
-- INSERT --                                                    12,1          All
```


File Edit View Terminal Tabs Help
- name: mongo
image: mongo:4.2
args:
- --bind_ip
- 0.0.0.0
ports:
- containerPort: 27017
resources:
requests:
memory: "160Mi"
limits:
memory: "320Mi"
:wq


File Edit View Terminal Tabs Help
To: <any> (traffic not restricted by destination)
Policy Types: Ingress, Egress
Name: default-deny
Namespace: ckad00018
Created on: 2022-09-24 04:27:37 +0000 UTC
Labels: <none>
Annotations: <none>
Spec:
PodSelector: <none> (Allowing the specific traffic to all pods in this namespace)
Allowing ingress traffic:
<none> (Selected pods are isolated for ingress connectivity)
Not affecting egress traffic
Policy Types: Ingress
candidate@node-1:~$ kubectl label pod ckad00018-newpod -n ckad00018 web-access=true
pod/ckad00018-newpod labeled
candidate@node-1:~$ kubectl label pod ckad00018-newpod -n ckad00018 db-access=true
pod/ckad00018-newpod labeled
candidate@node-1:~$ kubectl config use-context k8s
Switched to context "k8s".
candidate@node-1:~$ vim ~/chief-cardinal/nosql.yaml
candidate@node-1:~$ vim ~/chief-cardinal/nosql.yaml
candidate@node-1:~$ kubectl apply -f ~/chief-cardinal/nosql.yaml
deployment.apps/nosql configured
candidate@node-1:~$ kubectl get pods -n crayfish
NAME READY STATUS RESTARTS AGE
nosql-74cccf7d64-lkqlg 1/1 Running 0 3m2s
candidate@node-1:~$ kubectl get deploy -n crayfish
NAME READY UP-TO-DATE AVAILABLE AGE
nosql 1/1 1 1 7h16m
candidate@node-1:~$

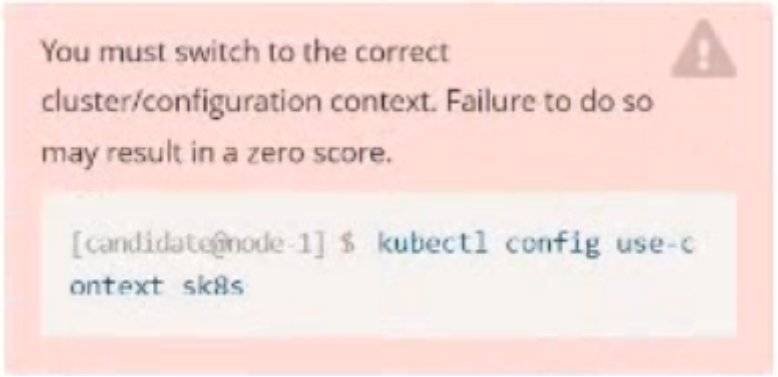
Answer:

A

Question 3

Question Type: MultipleChoice

Refer to Exhibit.

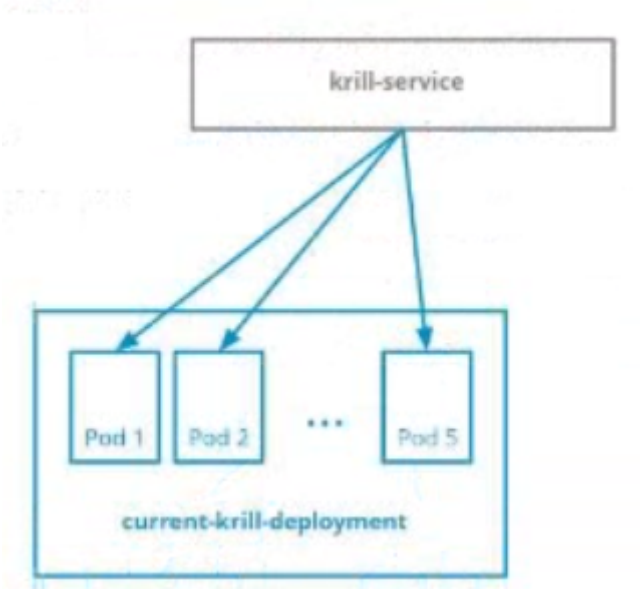


Context

You are asked to prepare a Canary deployment for testing a new application release.

Task:

A Service named krill-Service in the goshark namespace points to 5 pod created by the Deployment named current-krill-deployment

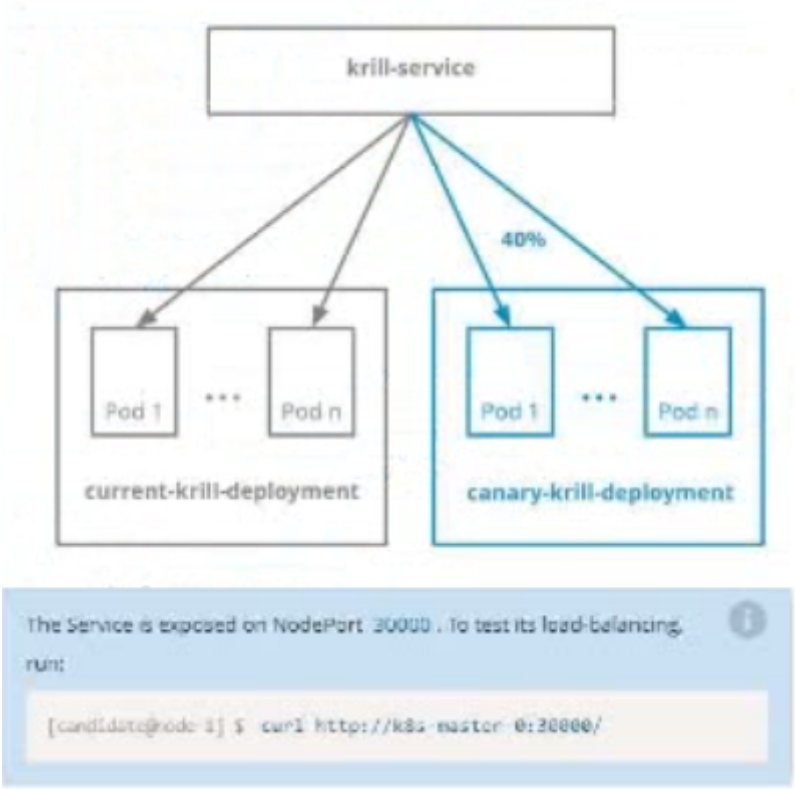


1) Create an identical Deployment named canary-kill-deployment, in the same namespace.

2) Modify the Deployment so that:

-A maximum number of 10 pods run in the goshawk namespace.

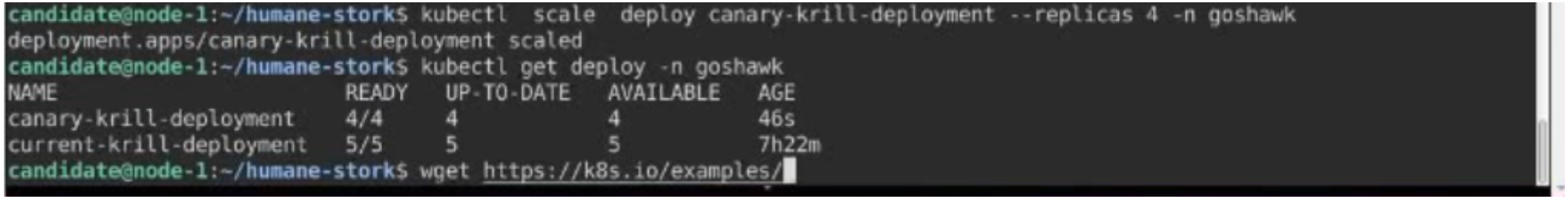
-40% of the krill-service 's traffic goes to the canary-krill-deployment pod(s)



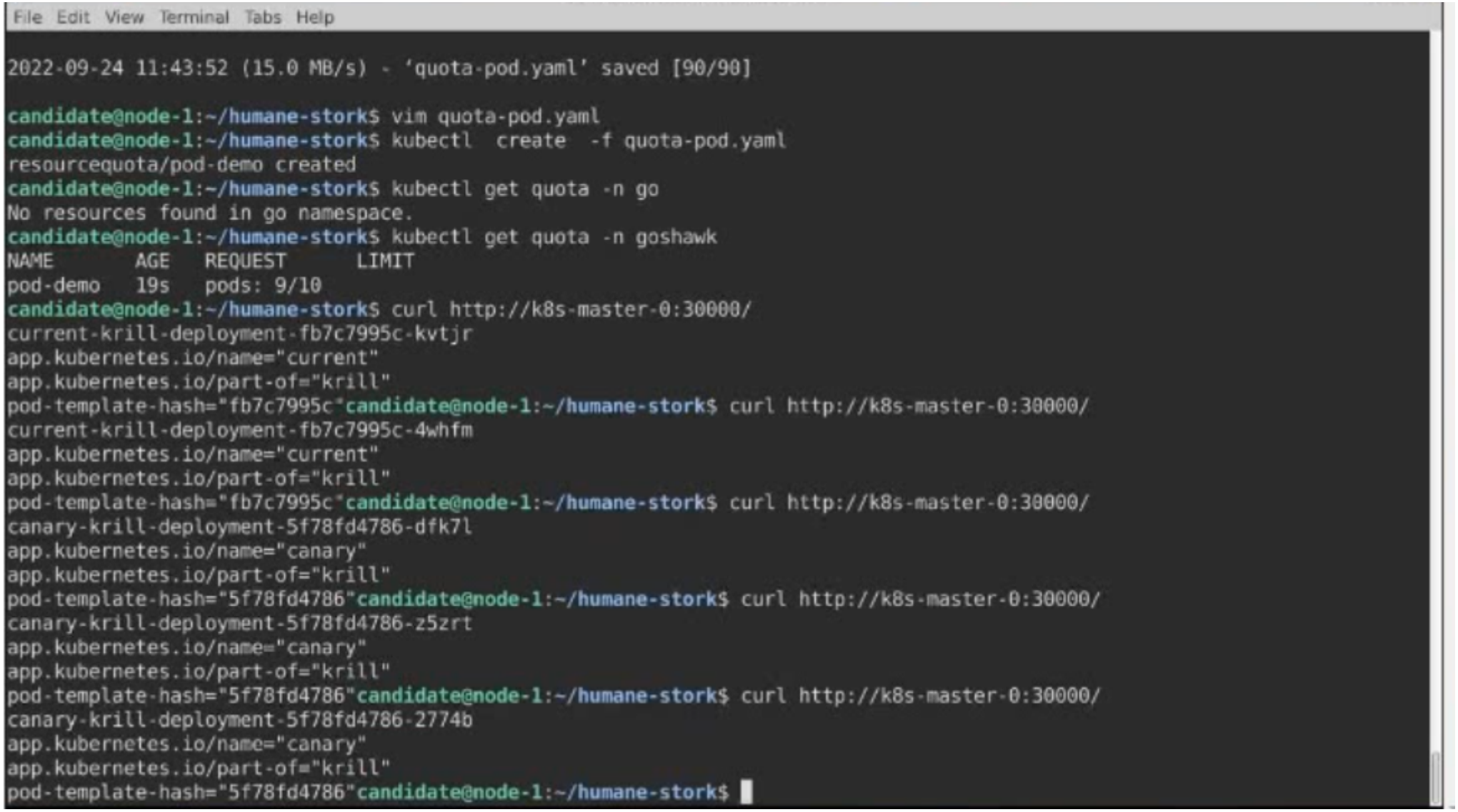
Options:

A- Explanation:

Solution:


candidate@node-1:~/humane-stork$ kubectl scale deploy canary-krill-deployment --replicas 4 -n goshawk
deployment.apps/canary-krill-deployment scaled
candidate@node-1:~/humane-stork$ kubectl get deploy -n goshawk
NAME READY UP-TO-DATE AVAILABLE AGE
canary-krill-deployment 4/4 4 4 46s
current-krill-deployment 5/5 5 5 7h22m
candidate@node-1:~/humane-stork$ wget https://k8s.io/examples/


File Edit View Terminal Tabs Help
candidate@node-1:~/humane-stork$ wget https://k8s.io/examples/admin/resource/quota-pod.yaml
--2022-09-24 11:43:51-- https://k8s.io/examples/admin/resource/quota-pod.yaml
Resolving k8s.io (k8s.io)... 34.107.204.206, 2600:1901:0:26f3::
Connecting to k8s.io (k8s.io)|34.107.204.206|:443... connected.
HTTP request sent, awaiting response... 301 Moved Permanently
Location: https://kubernetes.io/examples/admin/resource/quota-pod.yaml [following]
--2022-09-24 11:43:52-- https://kubernetes.io/examples/admin/resource/quota-pod.yaml
Resolving kubernetes.io (kubernetes.io)... 147.75.40.148
Connecting to kubernetes.io (kubernetes.io)|147.75.40.148|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 90 [application/x-yaml]
Saving to: 'quota-pod.yaml'
quota-pod.yaml 100%[=>] 90 --.-KB/s in 0s
2022-09-24 11:43:52 (15.0 MB/s) - 'quota-pod.yaml' saved [90/90]
candidate@node-1:~/humane-stork$ vim quota-pod.yaml


File Edit View Terminal Tabs Help
2022-09-24 11:43:52 (15.0 MB/s) - 'quota-pod.yaml' saved [90/90]
candidate@node-1:~/humane-stork$ vim quota-pod.yaml
candidate@node-1:~/humane-stork$ kubectl create -f quota-pod.yaml
resourcequota/pod-demo created
candidate@node-1:~/humane-stork$ kubectl get quota -n go
No resources found in go namespace.
candidate@node-1:~/humane-stork$ kubectl get quota -n goshawk
NAME AGE REQUEST LIMIT
pod-demo 19s pods: 9/10
candidate@node-1:~/humane-stork$ curl http://k8s-master-0:30000/
current-krill-deployment-fb7c7995c-kvtjr
app.kubernetes.io/name="current"
app.kubernetes.io/part-of="krill"
pod-template-hash="fb7c7995c"candidate@node-1:~/humane-stork$ curl http://k8s-master-0:30000/
current-krill-deployment-fb7c7995c-4whfm
app.kubernetes.io/name="current"
app.kubernetes.io/part-of="krill"
pod-template-hash="fb7c7995c"candidate@node-1:~/humane-stork$ curl http://k8s-master-0:30000/
canary-krill-deployment-5f78fd4786-dfk7l
app.kubernetes.io/name="canary"
app.kubernetes.io/part-of="krill"
pod-template-hash="5f78fd4786"candidate@node-1:~/humane-stork$ curl http://k8s-master-0:30000/
canary-krill-deployment-5f78fd4786-z5zrt
app.kubernetes.io/name="canary"
app.kubernetes.io/part-of="krill"
pod-template-hash="5f78fd4786"candidate@node-1:~/humane-stork$ curl http://k8s-master-0:30000/
canary-krill-deployment-5f78fd4786-2774b
app.kubernetes.io/name="canary"
app.kubernetes.io/part-of="krill"
pod-template-hash="5f78fd4786"candidate@node-1:~/humane-stork$

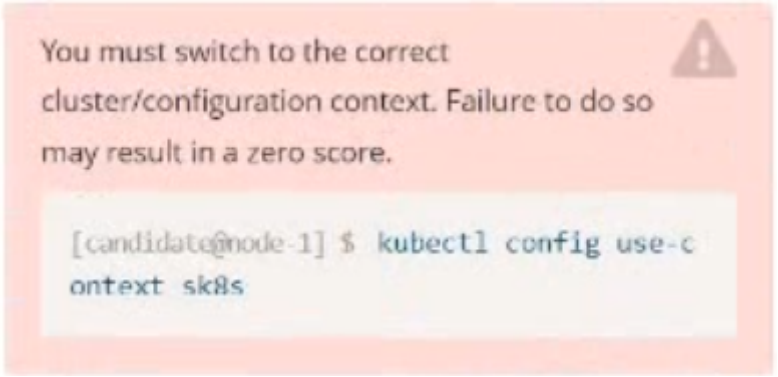
Answer:

A

Question 4

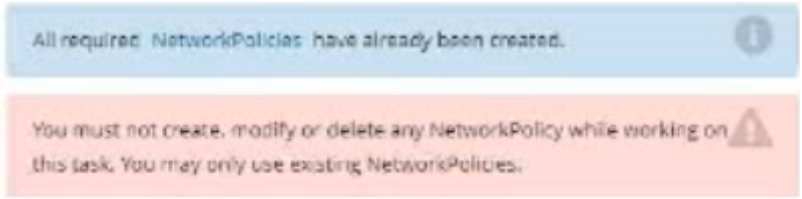
Question Type: MultipleChoice

Refer to Exhibit.



Task:

Update the Pod ckad00018-newpod in the ckad00018 namespace to use a NetworkPolicy allowing the Pod to send and receive traffic only to and from the pods web and db

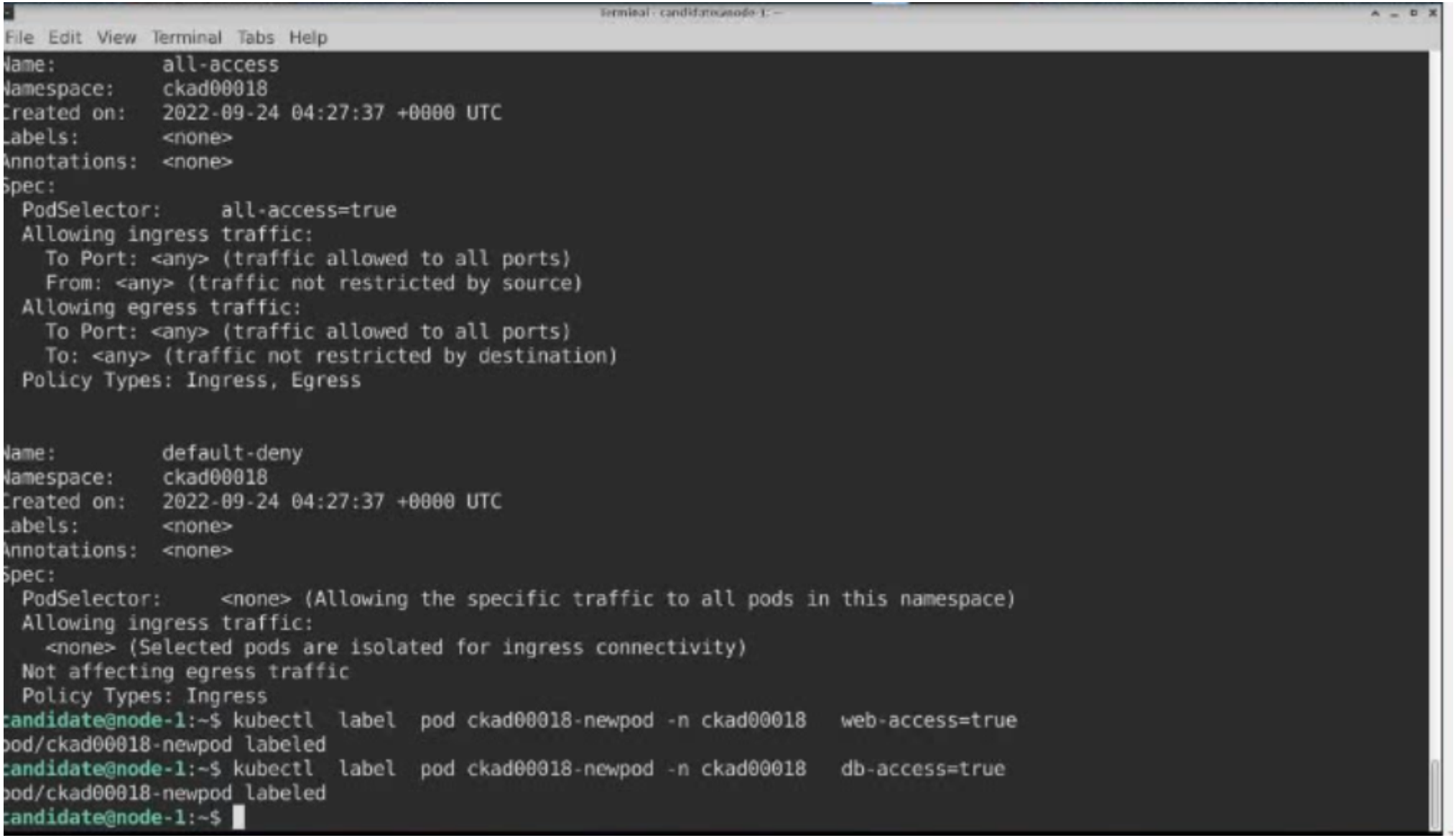


Options:

A- Explanation:

Solution:

```
candidate@node-1:~$ kubectl config use-context nk8s
Switched to context "nk8s".
candidate@node-1:~$ kubectl  describe  netpol -n ckad00018
```


File Edit View Terminal Tabs Help
Name: all-access
Namespace: ckad00018
Created on: 2022-09-24 04:27:37 +0000 UTC
Labels: <none>
Annotations: <none>
Spec:
PodSelector: all-access=true
Allowing ingress traffic:
To Port: <any> (traffic allowed to all ports)
From: <any> (traffic not restricted by source)
Allowing egress traffic:
To Port: <any> (traffic allowed to all ports)
To: <any> (traffic not restricted by destination)
Policy Types: Ingress, Egress
Name: default-deny
Namespace: ckad00018
Created on: 2022-09-24 04:27:37 +0000 UTC
Labels: <none>
Annotations: <none>
Spec:
PodSelector: <none> (Allowing the specific traffic to all pods in this namespace)
Allowing ingress traffic:
<none> (Selected pods are isolated for ingress connectivity)
Not affecting egress traffic
Policy Types: Ingress
candidate@node-1:~$ kubectl label pod ckad00018-newpod -n ckad00018 web-access=true
pod/ckad00018-newpod labeled
candidate@node-1:~$ kubectl label pod ckad00018-newpod -n ckad00018 db-access=true
pod/ckad00018-newpod labeled
candidate@node-1:~$

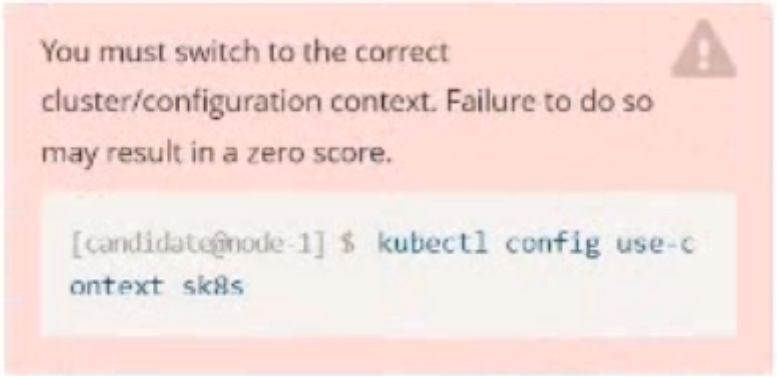
Answer:

A

Question 5

Question Type: MultipleChoice

Refer to Exhibit.



Task:

Modify the existing Deployment named broker-deployment running in namespace quetzal so that its containers.

1) Run with user ID 30000 and

2) Privilege escalation is forbidden

The broker-deployment is manifest file can be found at:

`~/daring-moccasin/broker-deployment.yaml`

Options:

A- Explanation:

Solution:

File Edit View Terminal Tabs Help

```yaml
    containers:
      - name: broker
        image: redis:alpine
        ports:
          - containerPort: 6379
        securityContext:
            runAsUser: 30000
            privileged: false
~
:wq
```

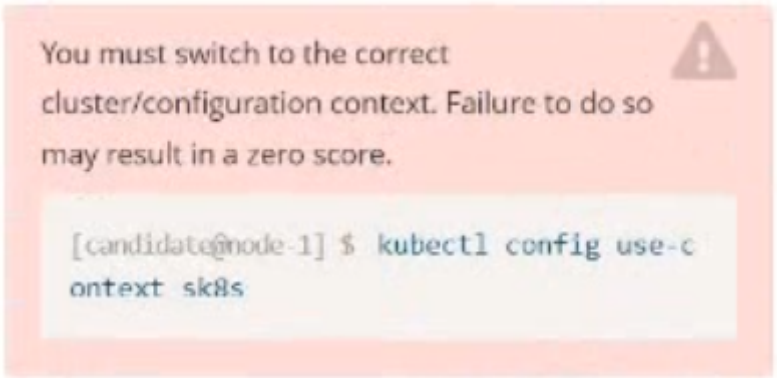
Answer:

A

Question 6

Question Type: MultipleChoice

Refer to Exhibit.



Task:

1- Update the Propertunel scaling configuration of the Deployment web1 in the ckad00015 namespace setting maxSurge to 2 and maxUnavailable to 59

2- Update the web1 Deployment to use version tag 1.13.7 for the Ifconf/nginx container image.

3- Perform a rollback of the web1 Deployment to its previous version

Options:

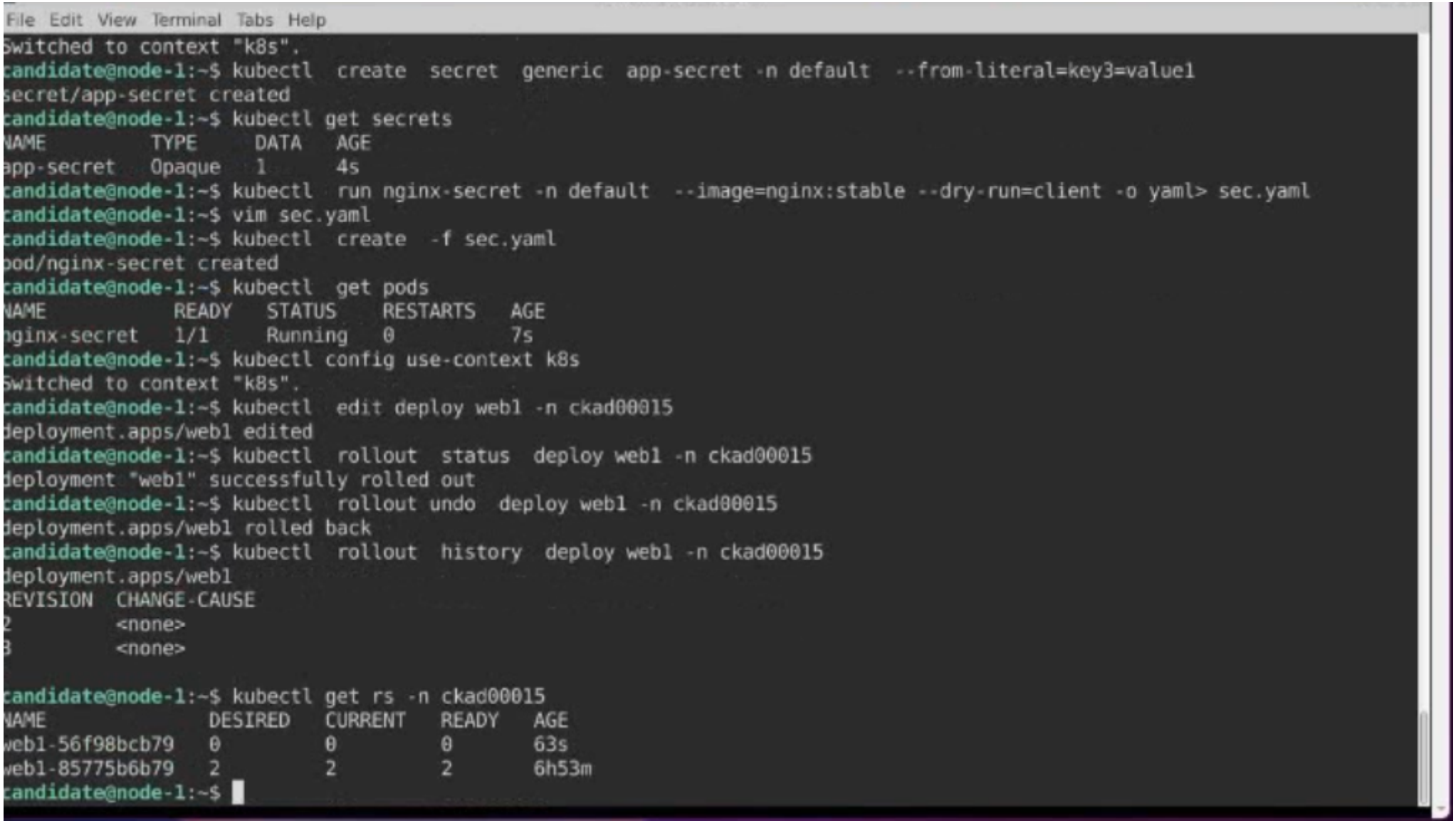
A- Explanation:

Solution:

```
candidate@node-1:~$ kubectl config use-context k8s
Switched to context "k8s".
candidate@node-1:~$ kubectl  edit deploy web1 -n ckad00015
```

File Edit View Terminal Tabs Help

```
      app: nginx
  strategy:
    rollingUpdate:
      maxSurge: 2%
      maxUnavailable: 5%
    type: RollingUpdate
  template:
    metadata:
      creationTimestamp: null
      labels:
        app: nginx
    spec:
      containers:
      - image: lfccncf/nginx:1.13.7
        imagePullPolicy: IfNotPresent
        name: nginx
        ports:
        - containerPort: 80
          protocol: TCP
        resources: {}
        terminationMessagePath: /dev/termination-log
        terminationMessagePolicy: File
      dnsPolicy: ClusterFirst
      restartPolicy: Always
      schedulerName: default-scheduler
      securityContext: {}
      terminationGracePeriodSeconds: 30
status:
  availableReplicas: 2
  conditions:
  - lastTransitionTime: "2022-09-24T04:26:41Z"
```


File Edit View Terminal Tabs Help
Switched to context "k8s".
candidate@node-1:~$ kubectl create secret generic app-secret -n default --from-literal=key3=value1
secret/app-secret created
candidate@node-1:~$ kubectl get secrets
NAME TYPE DATA AGE
app-secret Opaque 1 4s
candidate@node-1:~$ kubectl run nginx-secret -n default --image=nginx:stable --dry-run=client -o yaml> sec.yaml
candidate@node-1:~$ vim sec.yaml
candidate@node-1:~$ kubectl create -f sec.yaml
pod/nginx-secret created
candidate@node-1:~$ kubectl get pods
NAME READY STATUS RESTARTS AGE
nginx-secret 1/1 Running 0 7s
candidate@node-1:~$ kubectl config use-context k8s
Switched to context "k8s".
candidate@node-1:~$ kubectl edit deploy web1 -n ckad00015
deployment.apps/web1 edited
candidate@node-1:~$ kubectl rollout status deploy web1 -n ckad00015
deployment "web1" successfully rolled out
candidate@node-1:~$ kubectl rollout undo deploy web1 -n ckad00015
deployment.apps/web1 rolled back
candidate@node-1:~$ kubectl rollout history deploy web1 -n ckad00015
deployment.apps/web1
REVISION CHANGE-CAUSE
2 <none>
3 <none>
candidate@node-1:~$ kubectl get rs -n ckad00015
NAME DESIRED CURRENT READY AGE
web1-56f98bcb79 0 0 0 63s
web1-85775b6b79 2 2 2 6h53m
candidate@node-1:~$

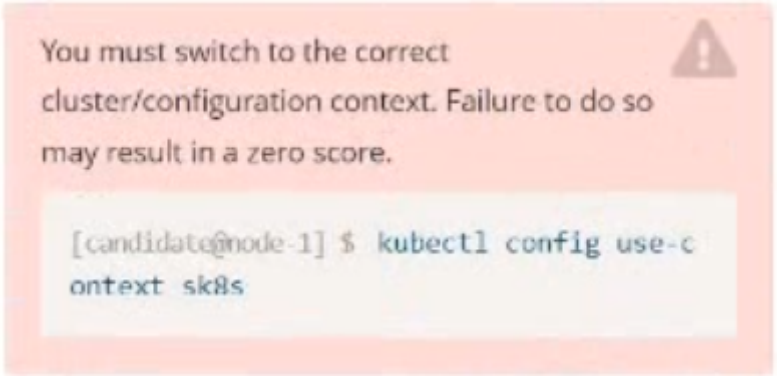
Answer:

A

Question 7

Question Type: MultipleChoice

Refer to Exhibit.



Task:

1) Create a secret named app-secret in the default namespace containing the following single key-value pair:

Key3: value1

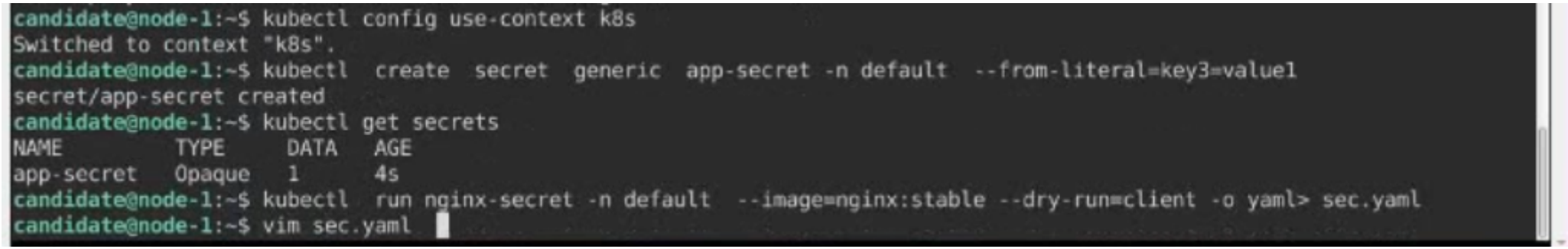
2) Create a Pod named ngnix secret in the default namespace.Specify a single container using the nginx:stable image.

Add an environment variable named BEST_VARIABLE consuming the value of the secret key3.

Options:

A- Explanation:

Solution:



File Edit View Terminal Tabs Help

```
apiVersion: v1
kind: Pod
metadata:
  creationTimestamp: null
  labels:
    run: nginx-secret
  name: nginx-secret
  namespace: default
spec:
  containers:
  - image: nginx:stable
    name: nginx-secret
    env:
       - name: BEST_VARIABLE
         valueFrom:
            secretKeyRef:
               name: app-secret
               key: key3
~
~
~
~
~
~
~
~
~
~
~
~
~
~
:wq
```

```
candidate@node-1:~$ kubectl config use-context k8s
Switched to context "k8s".
candidate@node-1:~$ kubectl  create  secret  generic  app-secret -n default  --from-literal=key3=value1
secret/app-secret created
candidate@node-1:~$ kubectl get secrets
NAME         TYPE     DATA   AGE
app-secret   Opaque   1      4s
candidate@node-1:~$ kubectl  run nginx-secret -n default  --image=nginx:stable --dry-run=client -o yaml> sec.yaml
candidate@node-1:~$ vim sec.yaml
candidate@node-1:~$ kubectl  create  -f sec.yaml
pod/nginx-secret created
candidate@node-1:~$ kubectl  get pods
NAME           READY   STATUS    RESTARTS   AGE
nginx-secret   1/1     Running   0          7s
candidate@node-1:~$
```

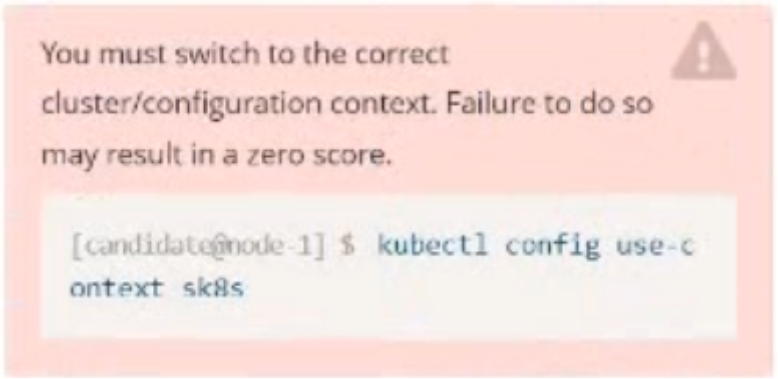
Answer:

A

Question 8

Question Type: MultipleChoice

Refer to Exhibit.



Task:

1) Fix any API depreciation issues in the manifest file -/credible-mite/www.yaml so that this application can be deployed on cluster K8s.

2) Deploy the application specified in the updated manifest file -/credible-mite/www.yaml in namespace cobra

Options:

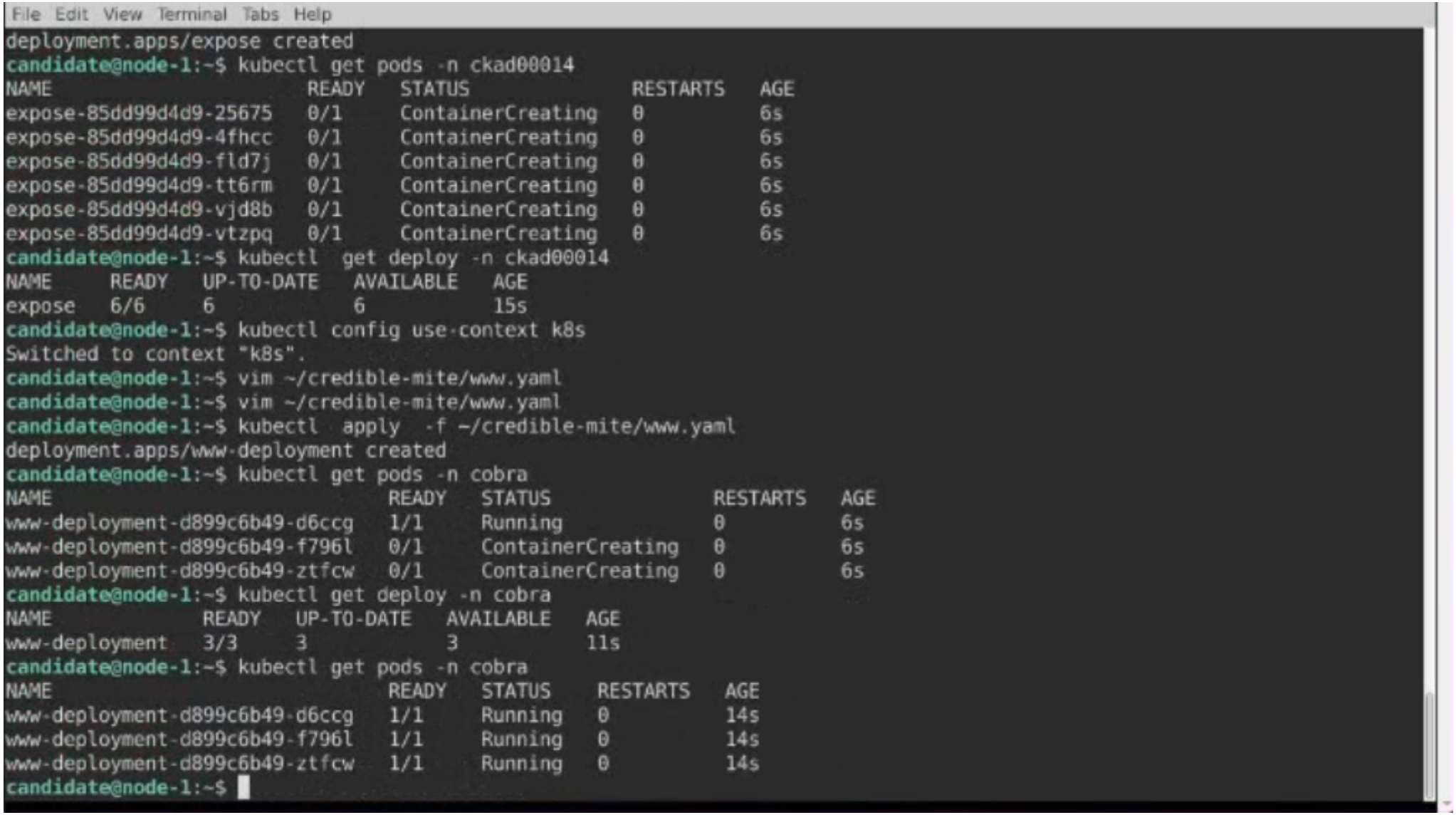
A- Explanation:

Solution:

```
candidate@node-1:~$ kubectl config use-context k8s
Switched to context "k8s".
candidate@node-1:~$ vim ~/credible-mite/www.yaml
```

File Edit View Terminal Tabs Help

```
apiVersion: apps/v1
kind: Deployment
metadata:
  name: www-deployment
  namespace: cobra
spec:
  replicas: 3
  selector:
        matchLabels:
              app: nginx
  template:
    metadata:
      labels:
        app: nginx
    spec:
      containers:
        - name: nginx
          image: "nginx:stable"
          ports:
            - containerPort: 80
          volumeMounts:
            - mountPath: /var/log/nginx
              name: logs
          env:
            - name: NGINX_ENTRYPOINT_QUIET_LOGS
              value: "1"
      volumes:
        - name: logs
          emptyDir: {}
~
~
:wq
```


File Edit View Terminal Tabs Help
deployment.apps/expose created
candidate@node-1:~$ kubectl get pods -n ckad00014
NAME READY STATUS RESTARTS AGE
expose-85dd99d4d9-25675 0/1 ContainerCreating 0 6s
expose-85dd99d4d9-4fhcc 0/1 ContainerCreating 0 6s
expose-85dd99d4d9-fld7j 0/1 ContainerCreating 0 6s
expose-85dd99d4d9-tt6rm 0/1 ContainerCreating 0 6s
expose-85dd99d4d9-vjd8b 0/1 ContainerCreating 0 6s
expose-85dd99d4d9-vtzpq 0/1 ContainerCreating 0 6s
candidate@node-1:~$ kubectl get deploy -n ckad00014
NAME READY UP-TO-DATE AVAILABLE AGE
expose 6/6 6 6 15s
candidate@node-1:~$ kubectl config use-context k8s
Switched to context "k8s".
candidate@node-1:~$ vim ~/credible-mite/www.yaml
candidate@node-1:~$ vim ~/credible-mite/www.yaml
candidate@node-1:~$ kubectl apply -f ~/credible-mite/www.yaml
deployment.apps/www-deployment created
candidate@node-1:~$ kubectl get pods -n cobra
NAME READY STATUS RESTARTS AGE
www-deployment-d899c6b49-d6ccg 1/1 Running 0 6s
www-deployment-d899c6b49-f796l 0/1 ContainerCreating 0 6s
www-deployment-d899c6b49-ztfcw 0/1 ContainerCreating 0 6s
candidate@node-1:~$ kubectl get deploy -n cobra
NAME READY UP-TO-DATE AVAILABLE AGE
www-deployment 3/3 3 3 11s
candidate@node-1:~$ kubectl get pods -n cobra
NAME READY STATUS RESTARTS AGE
www-deployment-d899c6b49-d6ccg 1/1 Running 0 14s
www-deployment-d899c6b49-f796l 1/1 Running 0 14s
www-deployment-d899c6b49-ztfcw 1/1 Running 0 14s
candidate@node-1:~$

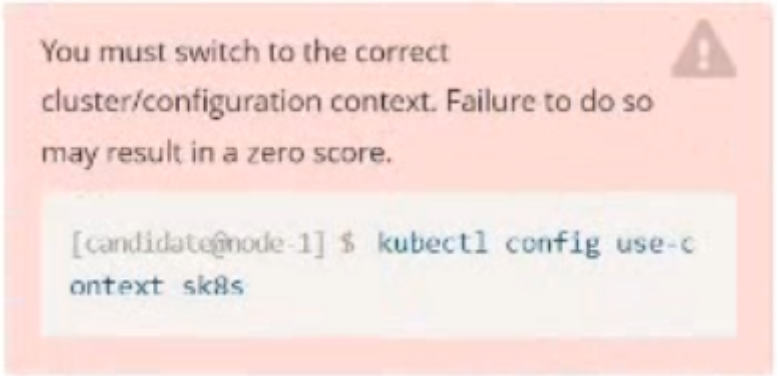
Answer:

A

Question 9

Question Type: MultipleChoice

Refer to Exhibit.



Task:

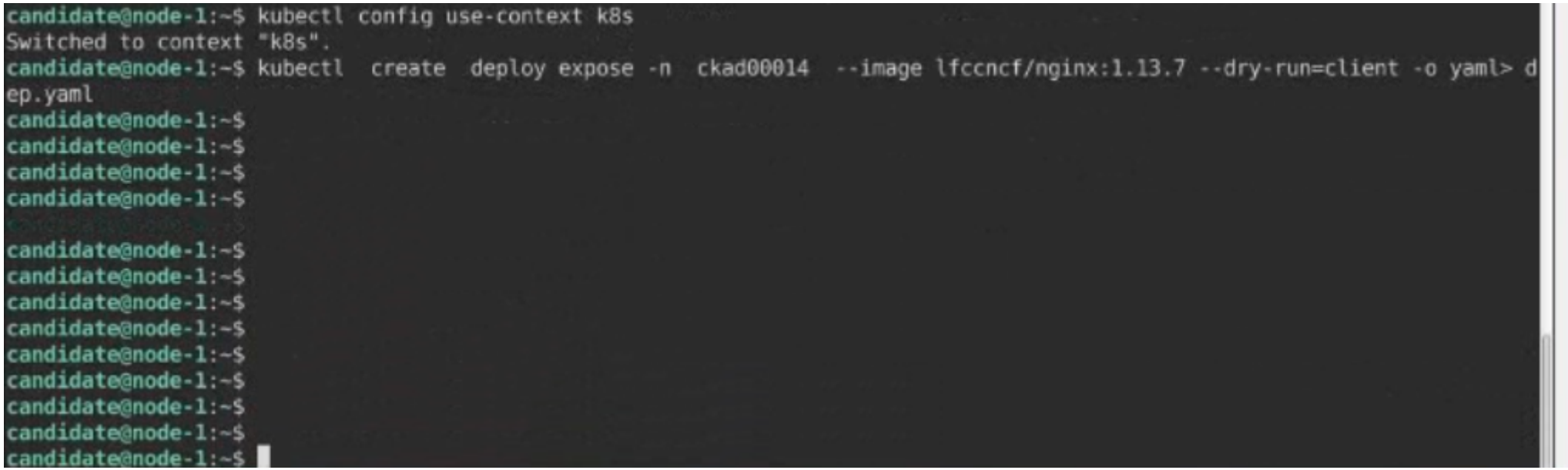
Create a Deployment named expose in the existing ckad00014 namespace running 6 replicas of a Pod. Specify a single container using the ifccncf/nginx: 1.13.7 image

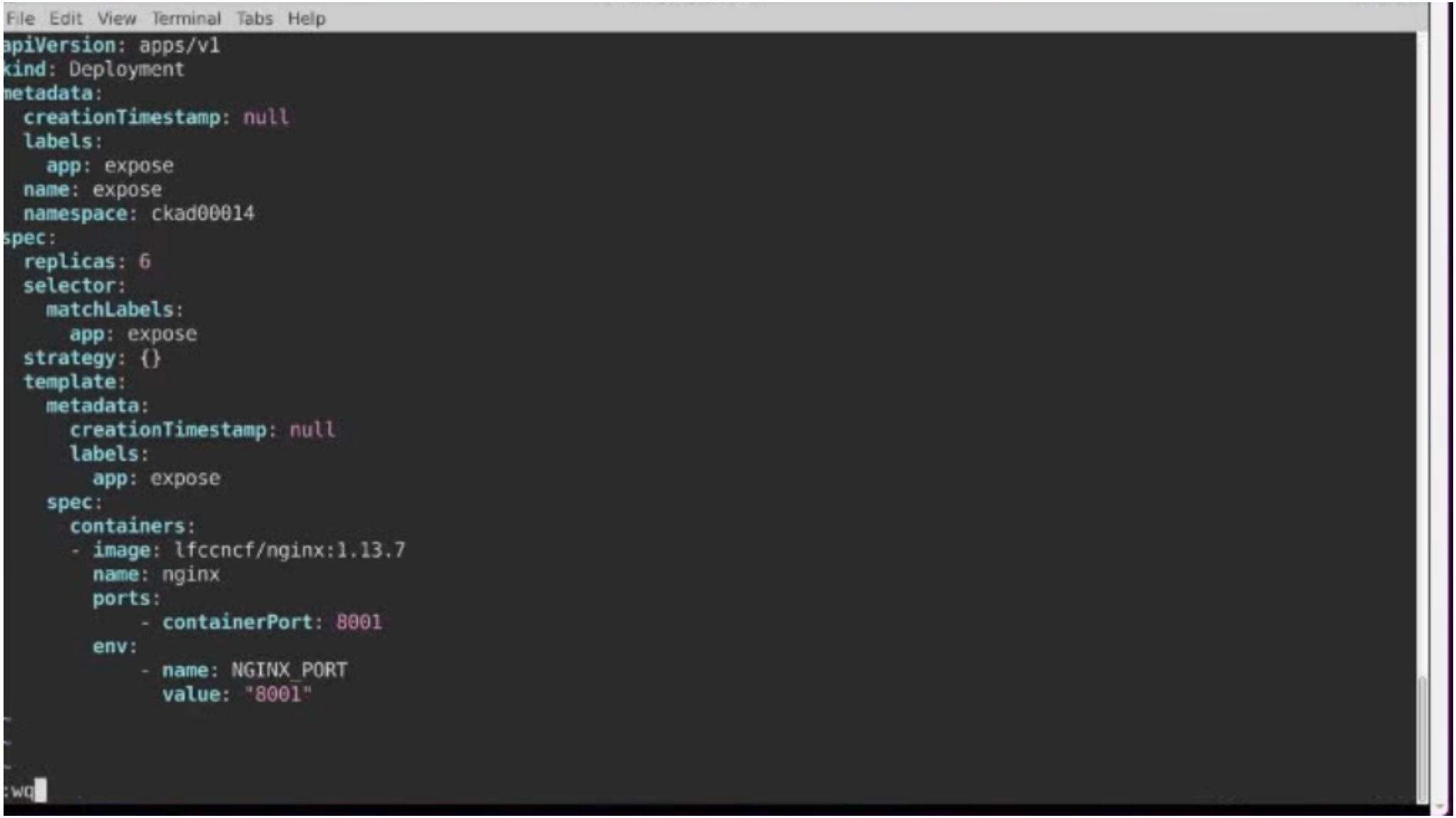
Add an environment variable named NGINX_PORT with the value 8001 to the container then expose port 8001

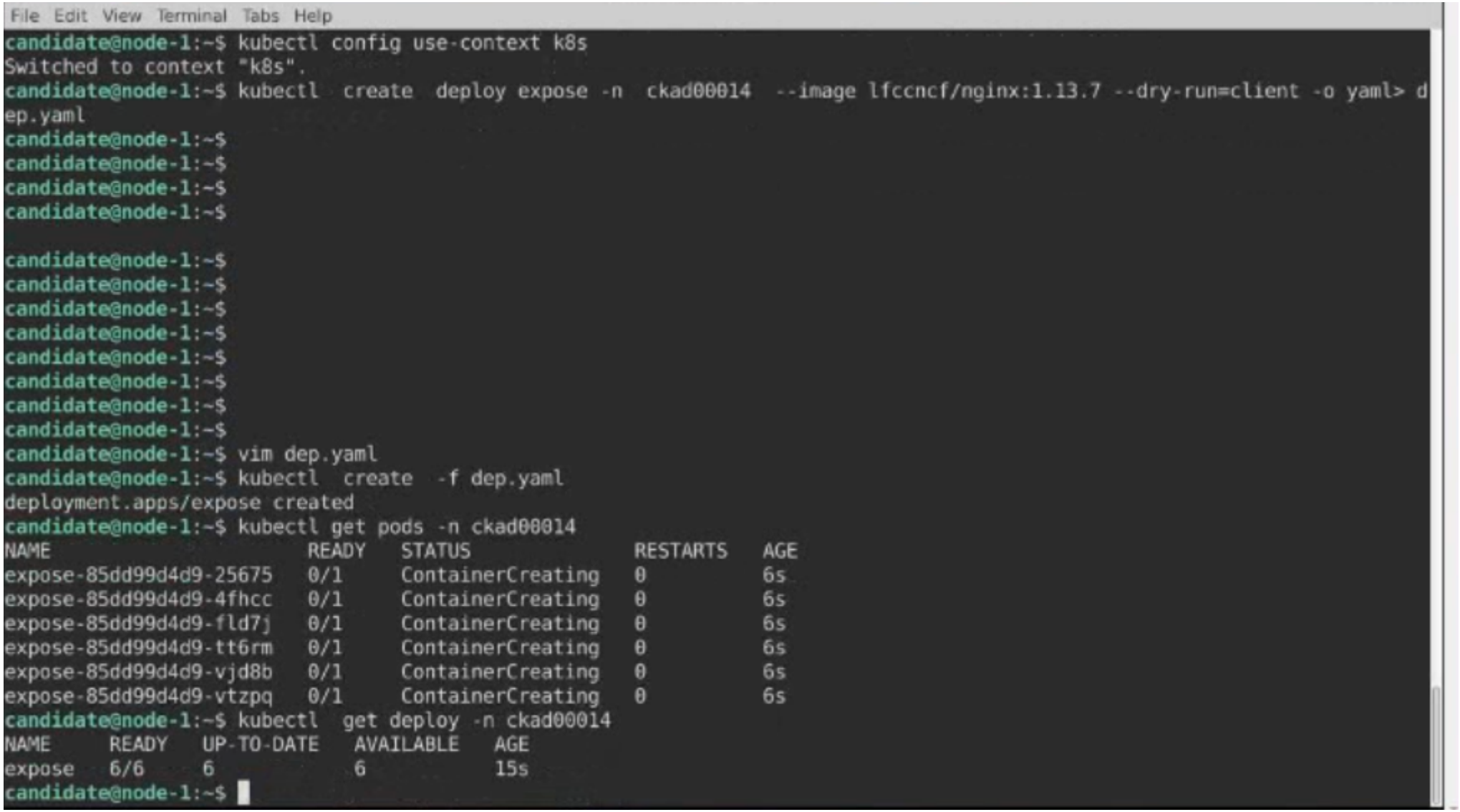
Options:

A- Explanation:

Solution:




File Edit View Terminal Tabs Help
apiVersion: apps/v1
kind: Deployment
metadata:
creationTimestamp: null
labels:
app: expose
name: expose
namespace: ckad00014
spec:
replicas: 6
selector:
matchLabels:
app: expose
strategy: {}
template:
metadata:
creationTimestamp: null
labels:
app: expose
spec:
containers:
- image: lfccncf/nginx:1.13.7
name: nginx
ports:
- containerPort: 8001
env:
- name: NGINX_PORT
value: "8001"
:wq


File Edit View Terminal Tabs Help
candidate@node-1:~$ kubectl config use-context k8s
Switched to context "k8s".
candidate@node-1:~$ kubectl create deploy expose -n ckad00014 --image lfccncf/nginx:1.13.7 --dry-run=client -o yaml> d
ep.yaml
candidate@node-1:~$
candidate@node-1:~$
candidate@node-1:~$
candidate@node-1:~$
candidate@node-1:~$
candidate@node-1:~$
candidate@node-1:~$
candidate@node-1:~$
candidate@node-1:~$
candidate@node-1:~$
candidate@node-1:~$
candidate@node-1:~$ vim dep.yaml
candidate@node-1:~$ kubectl create -f dep.yaml
deployment.apps/expose created
candidate@node-1:~$ kubectl get pods -n ckad00014
NAME READY STATUS RESTARTS AGE
expose-85dd99d4d9-25675 0/1 ContainerCreating 0 6s
expose-85dd99d4d9-4fhcc 0/1 ContainerCreating 0 6s
expose-85dd99d4d9-fld7j 0/1 ContainerCreating 0 6s
expose-85dd99d4d9-tt6rm 0/1 ContainerCreating 0 6s
expose-85dd99d4d9-vjd8b 0/1 ContainerCreating 0 6s
expose-85dd99d4d9-vtzpq 0/1 ContainerCreating 0 6s
candidate@node-1:~$ kubectl get deploy -n ckad00014
NAME READY UP-TO-DATE AVAILABLE AGE
expose 6/6 6 6 15s
candidate@node-1:~$

Answer:

A