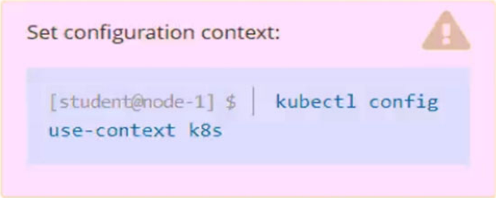
Certified Kubernetes Administrator v1.0 (CKA)

Page: 1 / 4
Total 20 questions

10 questions per page

Question 1 (Topic 1)

SIMULATION -


Set configuration context:

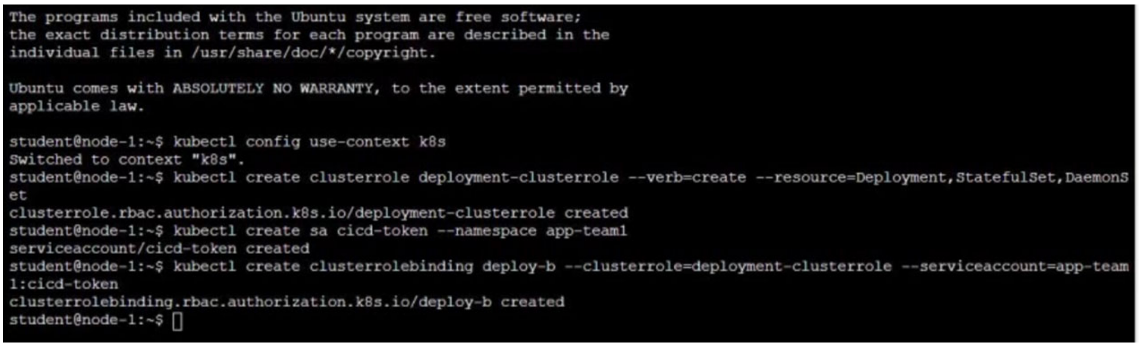
```
[student@node-1] $ | kubectl config
use-context k8s
```

Context -
You have been asked to create a new ClusterRole for a deployment pipeline and bind it to a specific ServiceAccount scoped to a specific namespace.

Task -
Create a new ClusterRole named deployment-clusterrole, which only allows to create the following resource types:
⇨ Deployment
⇨ Stateful Set
⇨ DaemonSet
Create a new ServiceAccount named cicd-token in the existing namespace app-team1.
Bind the new ClusterRole deployment-clusterrole to the new ServiceAccount cicd-token, limited to the namespace app-team1.

Answer : **See explanation below.**

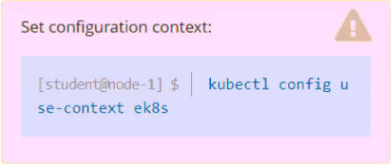
Explanation:



```
The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

student@node-1:~$ kubectl config use-context k8s
Switched to context "k8s".
student@node-1:~$ kubectl create clusterrole deployment-clusterrole --verb=create --resource=Deployment,StatefulSet,DaemonS
et
clusterrole.rbac.authorization.k8s.io/deployment-clusterrole created
student@node-1:~$ kubectl create sa cicd-token --namespace app-team1
serviceaccount/cicd-token created
student@node-1:~$ kubectl create clusterrolebinding deploy-b --clusterrole=deployment-clusterrole --serviceaccount=app-team
1:cicd-token
clusterrolebinding.rbac.authorization.k8s.io/deploy-b created
student@node-1:~$ ▯
```

Question 2 (Topic 1)

SIMULATION -

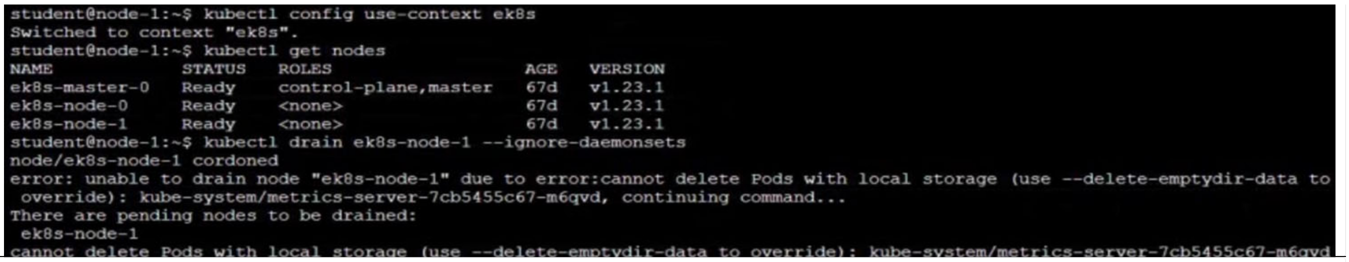

Set configuration context:

```
[student@node-1] $ | kubectl config u
se-context ek8s
```

Task -
Set the node named ek8s-node-0 as unavailable and reschedule all the pods running on it.

Answer : **See explanation below.**

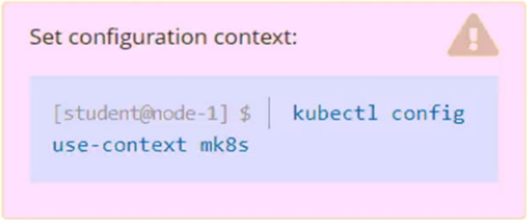
Explanation:



```
student@node-1:~$ kubectl config use-context ek8s
Switched to context "ek8s".
student@node-1:~$ kubectl get nodes
NAME             STATUS   ROLES                  AGE   VERSION
ek8s-master-0    Ready    control-plane,master   67d   v1.23.1
ek8s-node-0      Ready    <none>                 67d   v1.23.1
ek8s-node-1      Ready    <none>                 67d   v1.23.1
student@node-1:~$ kubectl drain ek8s-node-1 --ignore-daemonsets
node/ek8s-node-1 cordoned
error: unable to drain node "ek8s-node-1" due to error:cannot delete Pods with local storage (use --delete-emptydir-data to
 override): kube-system/metrics-server-7cb5455c67-m6qvd, continuing command...
There are pending nodes to be drained:
 ek8s-node-1
cannot delete Pods with local storage (use --delete-emptydir-data to override): kube-system/metrics-server-7cb5455c67-m6qvd
```

Question 3 (Topic 1)

SIMULATION -



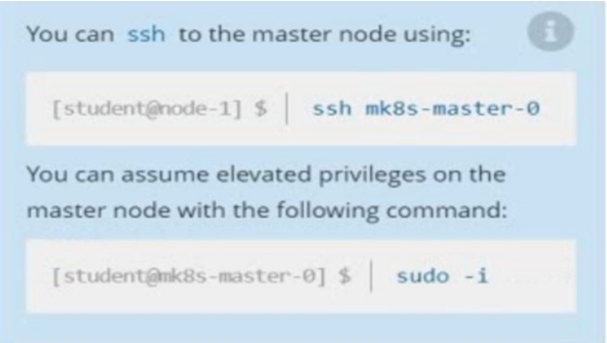
Set configuration context:

```
[student@node-1] $ | kubectl config
use-context mk8s
```

Task -

Given an existing Kubernetes cluster running version 1.22.1, upgrade all of the Kubernetes control plane and node components on the master node only to version 1.22.2.

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



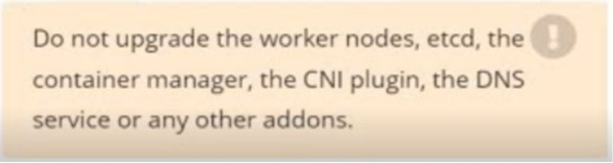
You can ssh to the master node using:

```
[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
```

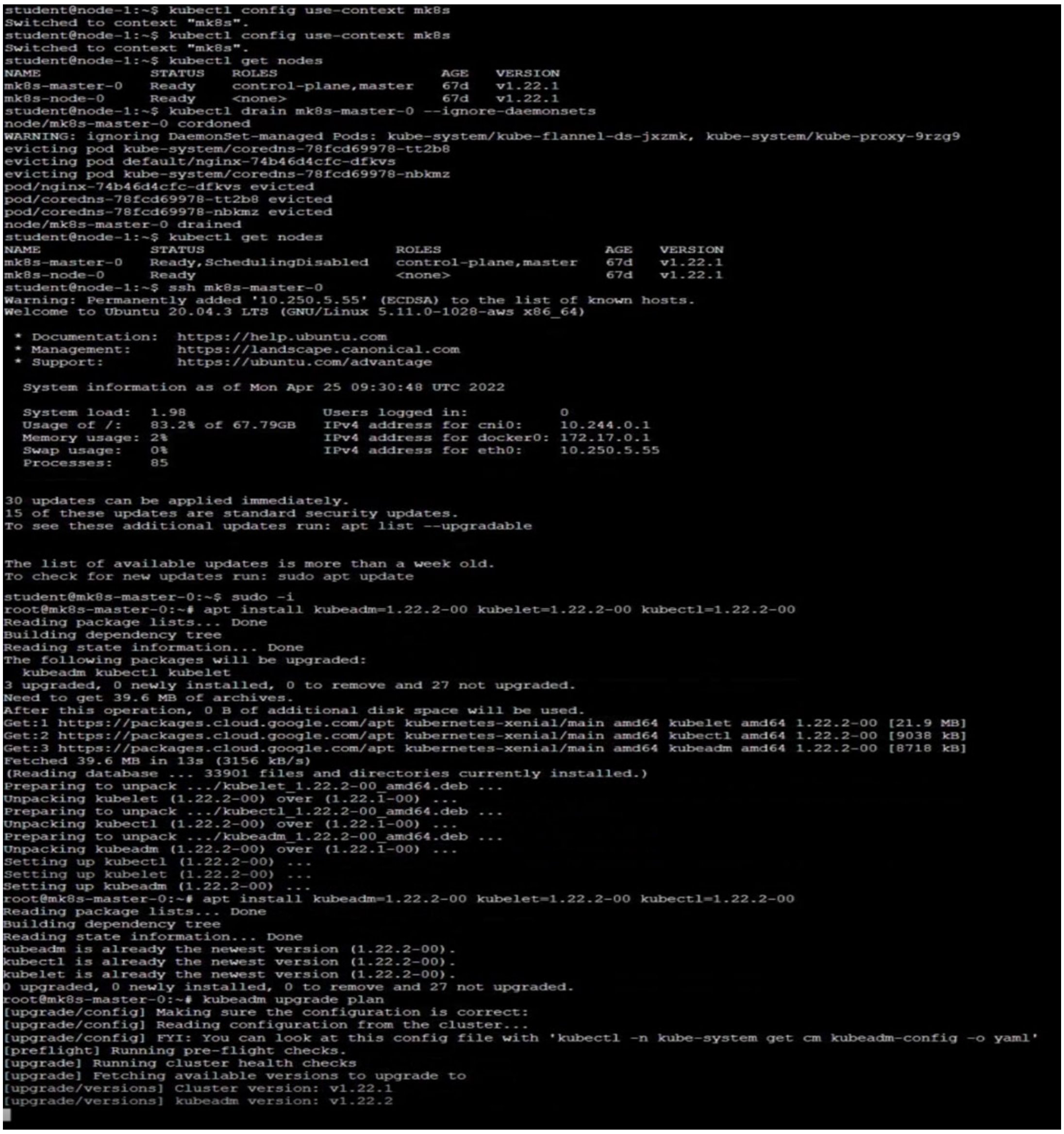
You are also expected to upgrade kubelet and kubectl on the master node.



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

Answer : **See explanation below.**

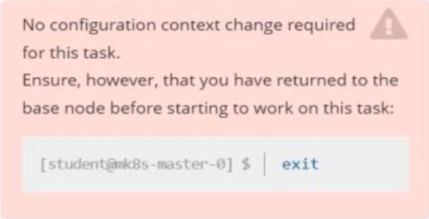
Explanation:



```
student@node-1:~$ kubectl config use-context mk8s
Switched to context "mk8s".
student@node-1:~$ kubectl config use-context mk8s
Switched to context "mk8s".
student@node-1:~$ kubectl get nodes
NAME            STATUS   ROLES                  AGE   VERSION
mk8s-master-0   Ready    control-plane,master   67d   v1.22.1
mk8s-node-0     Ready    <none>                 67d   v1.22.1
student@node-1:~$ kubectl drain mk8s-master-0 --ignore-daemonsets
node/mk8s-master-0 cordoned
WARNING: ignoring DaemonSet-managed Pods: kube-system/kube-flannel-ds-jxzmk, kube-system/kube-proxy-9rzg9
evicting pod kube-system/coredns-78fcd69978-tt2b8
evicting pod default/nginx-74b46d4cfc-dfkvs
evicting pod kube-system/coredns-78fcd69978-nbkmz
pod/nginx-74b46d4cfc-dfkvs evicted
pod/coredns-78fcd69978-tt2b8 evicted
pod/coredns-78fcd69978-nbkmz evicted
node/mk8s-master-0 drained
student@node-1:~$ kubectl get nodes
NAME            STATUS                     ROLES                  AGE   VERSION
mk8s-master-0   Ready,SchedulingDisabled   control-plane,master   67d   v1.22.1
mk8s-node-0     Ready                      <none>                 67d   v1.22.1
student@node-1:~$ ssh mk8s-master-0
Warning: Permanently added '10.250.5.55' (ECDSA) to the list of known hosts.
Welcome to Ubuntu 20.04.3 LTS (GNU/Linux 5.11.0-1028-aws x86_64)

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

  System information as of Mon Apr 25 09:30:48 UTC 2022

  System load:  1.98               Users logged in:          0
  Usage of /:   83.2% of 67.79GB   IPv4 address for cni0:    10.244.0.1
  Memory usage: 2%                 IPv4 address for docker0: 172.17.0.1
  Swap usage:   0%                 IPv4 address for eth0:    10.250.5.55
  Processes:    85


30 updates can be applied immediately.
15 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable


The list of available updates is more than a week old.
To check for new updates run: sudo apt update

student@mk8s-master-0:~$ sudo -i
root@mk8s-master-0:~# apt install kubeadm=1.22.2-00 kubelet=1.22.2-00 kubectl=1.22.2-00
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following packages will be upgraded:
  kubeadm kubectl kubelet
3 upgraded, 0 newly installed, 0 to remove and 27 not upgraded.
Need to get 39.6 MB of archives.
After this operation, 0 B of additional disk space will be used.
Get:1 https://packages.cloud.google.com/apt kubernetes-xenial/main amd64 kubelet amd64 1.22.2-00 [21.9 MB]
Get:2 https://packages.cloud.google.com/apt kubernetes-xenial/main amd64 kubectl amd64 1.22.2-00 [9038 kB]
Get:3 https://packages.cloud.google.com/apt kubernetes-xenial/main amd64 kubeadm amd64 1.22.2-00 [8718 kB]
Fetched 39.6 MB in 13s (3156 kB/s)
(Reading database ... 33901 files and directories currently installed.)
Preparing to unpack .../kubelet_1.22.2-00_amd64.deb ...
Unpacking kubelet (1.22.2-00) over (1.22.1-00) ...
Preparing to unpack .../kubectl_1.22.2-00_amd64.deb ...
Unpacking kubectl (1.22.2-00) over (1.22.1-00) ...
Preparing to unpack .../kubeadm_1.22.2-00_amd64.deb ...
Unpacking kubeadm (1.22.2-00) over (1.22.1-00) ...
Setting up kubectl (1.22.2-00) ...
Setting up kubelet (1.22.2-00) ...
Setting up kubeadm (1.22.2-00) ...
root@mk8s-master-0:~# apt install kubeadm=1.22.2-00 kubelet=1.22.2-00 kubectl=1.22.2-00
Reading package lists... Done
Building dependency tree
Reading state information... Done
kubeadm is already the newest version (1.22.2-00).
kubectl is already the newest version (1.22.2-00).
kubelet is already the newest version (1.22.2-00).
0 upgraded, 0 newly installed, 0 to remove and 27 not upgraded.
root@mk8s-master-0:~# kubeadm upgrade plan
[upgrade/config] Making sure the configuration is correct:
[upgrade/config] Reading configuration from the cluster...
[upgrade/config] FYI: You can look at this config file with 'kubectl -n kube-system get cm kubeadm-config -o yaml'
[preflight] Running pre-flight checks.
[upgrade] Running cluster health checks
[upgrade] Fetching available versions to upgrade to
[upgrade/versions] Cluster version: v1.22.1
[upgrade/versions] kubeadm version: v1.22.2
```

Question 4 (Topic 1)

SIMULATION -



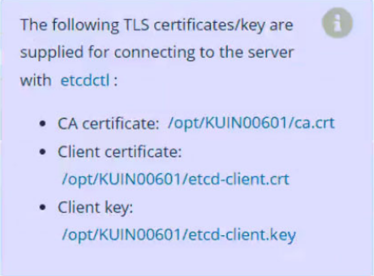
No configuration context change required for this task.

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

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

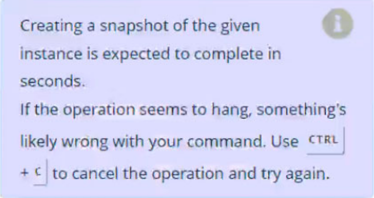
Task -

First, create a snapshot of the existing etcd instance running at https://127.0.0.1:2379, saving the snapshot to /var/lib/backup/etcd-snapshot.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



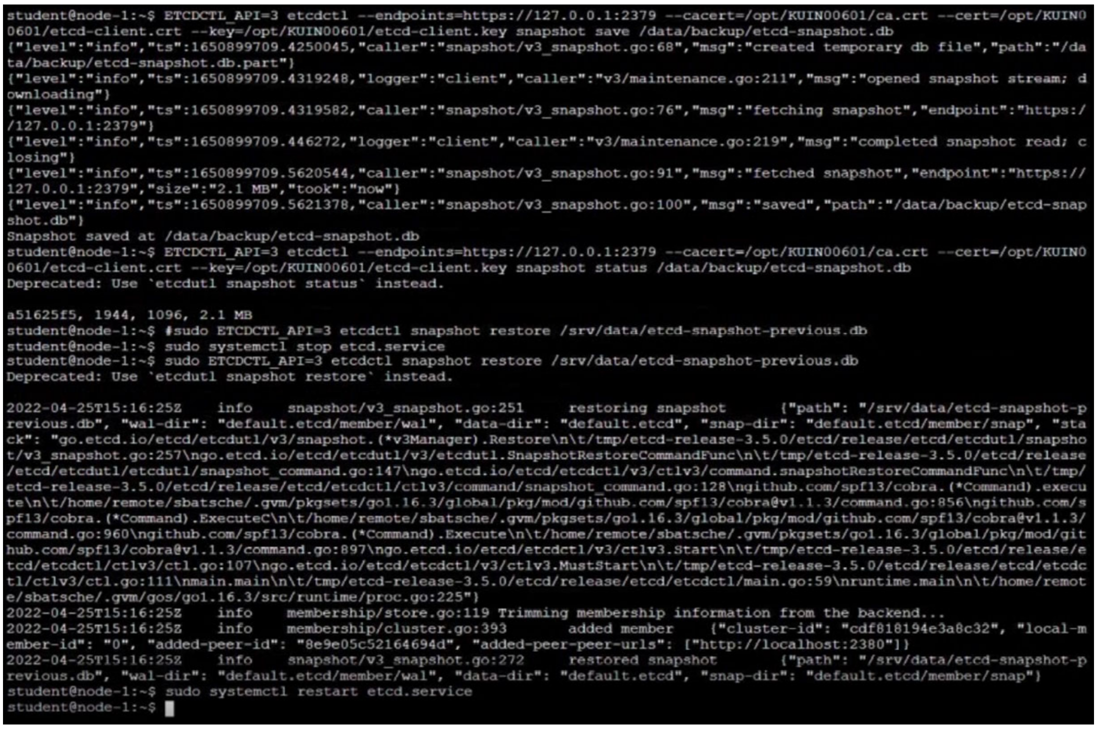
Creating a snapshot of the given instance is expected to complete in seconds.

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.

Answer : **See explanation below.**

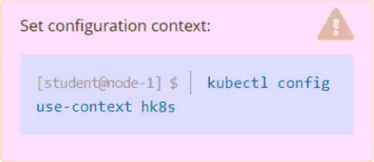
Explanation:



```
student@node-1:~$ ETCDCTL_API=3 etcdctl --endpoints=https://127.0.0.1:2379 --cacert=/opt/KUIN00601/ca.crt --cert=/opt/KUIN0
0601/etcd-client.crt --key=/opt/KUIN00601/etcd-client.key snapshot save /data/backup/etcd-snapshot.db
{"level":"info","ts":1650899709.4250045,"caller":"snapshot/v3_snapshot.go:68","msg":"created temporary db file","path":"/da
ta/backup/etcd-snapshot.db.part"}
{"level":"info","ts":1650899709.4319248,"logger":"client","caller":"v3/maintenance.go:211","msg":"opened snapshot stream; d
ownloading"}
{"level":"info","ts":1650899709.4319582,"caller":"snapshot/v3_snapshot.go:76","msg":"fetching snapshot","endpoint":"https:/
/127.0.0.1:2379"}
{"level":"info","ts":1650899709.446272,"logger":"client","caller":"v3/maintenance.go:219","msg":"completed snapshot read; c
losing"}
{"level":"info","ts":1650899709.5620544,"caller":"snapshot/v3_snapshot.go:91","msg":"fetched snapshot","endpoint":"https://
127.0.0.1:2379","size":"2.1 MB","took":"now"}
{"level":"info","ts":1650899709.5621378,"caller":"snapshot/v3_snapshot.go:100","msg":"saved","path":"/data/backup/etcd-snap
shot.db"}
Snapshot saved at /data/backup/etcd-snapshot.db
student@node-1:~$ ETCDCTL_API=3 etcdctl --endpoints=https://127.0.0.1:2379 --cacert=/opt/KUIN00601/ca.crt --cert=/opt/KUIN0
0601/etcd-client.crt --key=/opt/KUIN00601/etcd-client.key snapshot status /data/backup/etcd-snapshot.db
Deprecated: Use `etcdutl snapshot status` instead.

a51625f5, 1944, 1096, 2.1 MB
student@node-1:~$ #sudo ETCDCTL_API=3 etcdctl snapshot restore /srv/data/etcd-snapshot-previous.db
student@node-1:~$ sudo systemctl stop etcd.service
student@node-1:~$ sudo ETCDCTL_API=3 etcdctl snapshot restore /srv/data/etcd-snapshot-previous.db
Deprecated: Use `etcdutl snapshot restore` instead.

2022-04-25T15:16:25Z    info    snapshot/v3_snapshot.go:251     restoring snapshot      {"path": "/srv/data/etcd-snapshot-p
revious.db", "wal-dir": "default.etcd/member/wal", "data-dir": "default.etcd", "snap-dir": "default.etcd/member/snap", "sta
ck": "go.etcd.io/etcd/etcdutl/v3/snapshot.(*v3Manager).Restore\n\t/tmp/etcd-release-3.5.0/etcd/release/etcd/etcdutl/snapsho
t/v3_snapshot.go:257\ngo.etcd.io/etcd/etcdutl/v3/etcdutl.SnapshotRestoreCommandFunc\n\t/tmp/etcd-release-3.5.0/etcd/release
/etcd/etcdutl/etcdutl/snapshot_command.go:147\ngo.etcd.io/etcd/etcdctl/v3/ctlv3/command.snapshotRestoreCommandFunc\n\t/tmp/
etcd-release-3.5.0/etcd/release/etcd/etcdctl/ctlv3/command/snapshot_command.go:128\ngithub.com/spf13/cobra.(*Command).execu
te\n\t/home/remote/sbatsche/.gvm/pkgsets/go1.16.3/global/pkg/mod/github.com/spf13/cobra@v1.1.3/command.go:856\ngithub.com/s
pf13/cobra.(*Command).ExecuteC\n\t/home/remote/sbatsche/.gvm/pkgsets/go1.16.3/global/pkg/mod/github.com/spf13/cobra@v1.1.3/
command.go:960\ngithub.com/spf13/cobra.(*Command).Execute\n\t/home/remote/sbatsche/.gvm/pkgsets/go1.16.3/global/pkg/mod/git
hub.com/spf13/cobra@v1.1.3/command.go:897\ngo.etcd.io/etcd/etcdctl/v3/ctlv3.Start\n\t/tmp/etcd-release-3.5.0/etcd/release/e
tcd/etcdctl/ctlv3/ctl.go:107\ngo.etcd.io/etcd/etcdctl/v3/ctlv3.MustStart\n\t/tmp/etcd-release-3.5.0/etcd/release/etcd/etcdc
tl/ctlv3/ctl.go:111\nmain.main\n\t/tmp/etcd-release-3.5.0/etcd/release/etcd/etcdctl/main.go:59\nruntime.main\n\t/home/remot
e/sbatsche/.gvm/gos/go1.16.3/src/runtime/proc.go:225"}
2022-04-25T15:16:25Z    info    membership/store.go:119 Trimming membership information from the backend...
2022-04-25T15:16:25Z    info    membership/cluster.go:393       added member    {"cluster-id": "cdf818194e3a8c32", "local-m
ember-id": "0", "added-peer-id": "8e9e05c52164694d", "added-peer-peer-urls": ["http://localhost:2380"]}
2022-04-25T15:16:25Z    info    snapshot/v3_snapshot.go:272     restored snapshot       {"path": "/srv/data/etcd-snapshot-p
revious.db", "wal-dir": "default.etcd/member/wal", "data-dir": "default.etcd", "snap-dir": "default.etcd/member/snap"}
student@node-1:~$ sudo systemctl restart etcd.service
student@node-1:~$ █
```

Question 5 (Topic 1)

SIMULATION -



Set configuration context:

```
[student@node-1] $ | kubectl config
use-context hk8s
```

Task -

Create a new NetworkPolicy named allow-port-from-namespace in the existing namespace fubar.

Ensure that the new NetworkPolicy allows Pods in namespace internal to connect to port 9000 of Pods in namespace fubar.

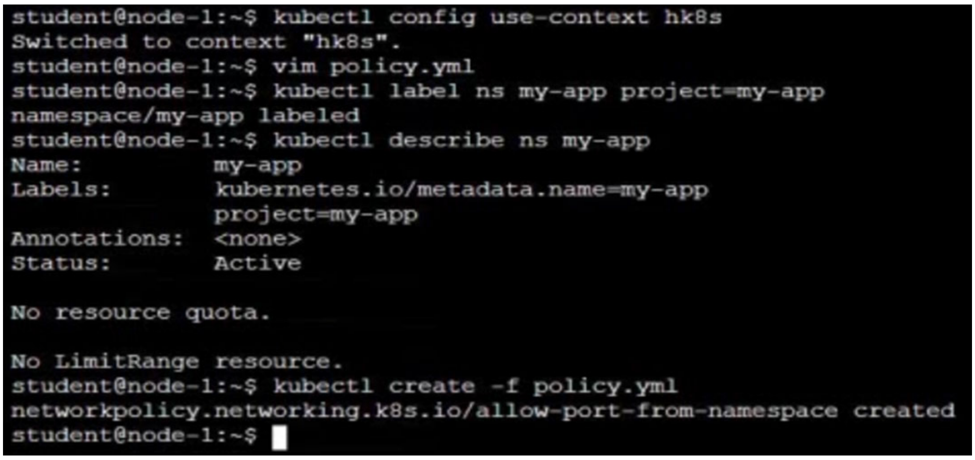
Further ensure that the new NetworkPolicy:

➾ does not allow access to Pods, which don't listen on port 9000

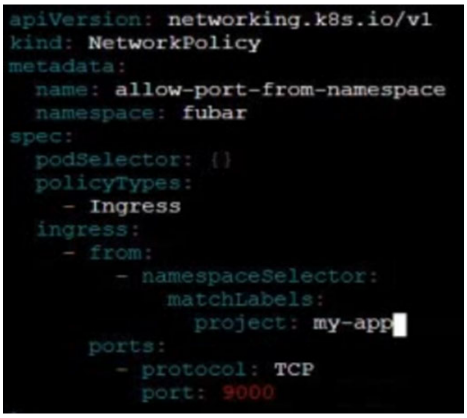
➾ does not allow access from Pods, which are not in namespace internal

Answer : **See explanation below.**

Explanation:



```
student@node-1:~$ kubectl config use-context hk8s
Switched to context "hk8s".
student@node-1:~$ vim policy.yml
student@node-1:~$ kubectl label ns my-app project=my-app
namespace/my-app labeled
student@node-1:~$ kubectl describe ns my-app
Name:          my-app
Labels:        kubernetes.io/metadata.name=my-app
               project=my-app
Annotations:   <none>
Status:        Active

No resource quota.

No LimitRange resource.
student@node-1:~$ kubectl create -f policy.yml
networkpolicy.networking.k8s.io/allow-port-from-namespace created
student@node-1:~$
```



```
apiVersion: networking.k8s.io/v1
kind: NetworkPolicy
metadata:
  name: allow-port-from-namespace
  namespace: fubar
spec:
  podSelector: {}
  policyTypes:
    - Ingress
  ingress:
    - from:
        - namespaceSelector:
            matchLabels:
              project: my-app
      ports:
        - protocol: TCP
          port: 9000
```

Page: 1 / 4
Total 20 questions
Next Page
10 questions per page

Certified Kubernetes Administrator v1.0 (CKA)

Page: 2 / 4
Total 20 questions

10 questions per page

Question 6 (Topic 1)

SIMULATION -

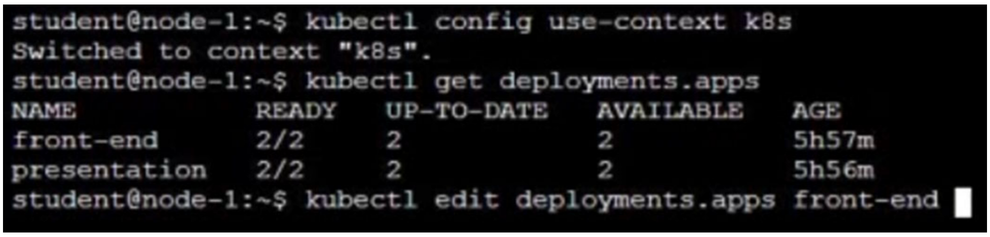
Set configuration context:

```
[student@node-1] $ | kubectl config use-context k8s
```

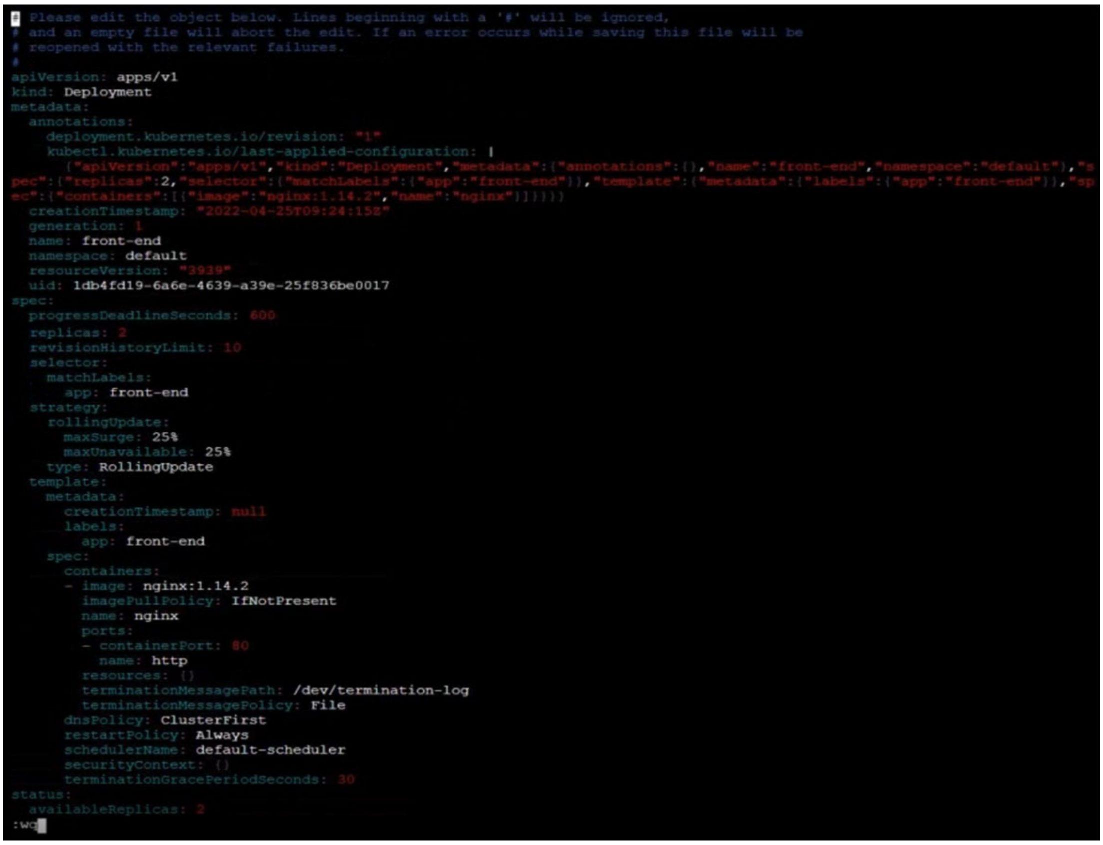
Task -
Reconfigure the existing deployment front-end and add a port specification named http exposing port 80/tcp of the existing container nginx.
Create a new service named front-end-svc exposing the container port http.
Configure the new service to also expose the individual Pods via a NodePort on the nodes on which they are scheduled.

Answer : **See explanation below.**

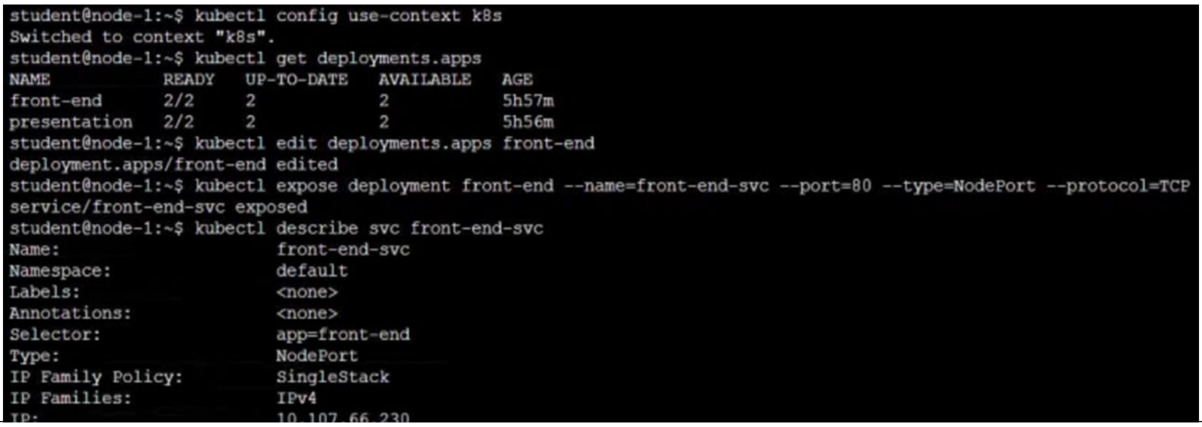
Explanation:



```
student@node-1:~$ kubectl config use-context k8s
Switched to context "k8s".
student@node-1:~$ kubectl get deployments.apps
NAME            READY   UP-TO-DATE   AVAILABLE   AGE
front-end       2/2     2            2           5h57m
presentation    2/2     2            2           5h56m
student@node-1:~$ kubectl edit deployments.apps front-end
```



```
# Please edit the object below. Lines beginning with a '#' will be ignored,
# and an empty file will abort the edit. If an error occurs while saving this file will be
# reopened with the relevant failures.
#
apiVersion: apps/v1
kind: Deployment
metadata:
  annotations:
    deployment.kubernetes.io/revision: "1"
    kubectl.kubernetes.io/last-applied-configuration: |
      {"apiVersion":"apps/v1","kind":"Deployment","metadata":{"annotations":{},"name":"front-end","namespace":"default"},"s
pec":{"replicas":2,"selector":{"matchLabels":{"app":"front-end"}},"template":{"metadata":{"labels":{"app":"front-end"}},"sp
ec":{"containers":[{"image":"nginx:1.14.2","name":"nginx"}]}}}}
  creationTimestamp: "2022-04-25T09:24:15Z"
  generation: 1
  name: front-end
  namespace: default
  resourceVersion: "3939"
  uid: 1db4fd19-6a6e-4639-a39e-25f836be0017
spec:
  progressDeadlineSeconds: 600
  replicas: 2
  revisionHistoryLimit: 10
  selector:
    matchLabels:
      app: front-end
  strategy:
    rollingUpdate:
      maxSurge: 25%
      maxUnavailable: 25%
    type: RollingUpdate
  template:
    metadata:
      creationTimestamp: null
      labels:
        app: front-end
    spec:
      containers:
      - image: nginx:1.14.2
        imagePullPolicy: IfNotPresent
        name: nginx
        ports:
        - containerPort: 80
          name: http
        resources: {}
        terminationMessagePath: /dev/termination-log
        terminationMessagePolicy: File
      dnsPolicy: ClusterFirst
      restartPolicy: Always
      schedulerName: default-scheduler
      securityContext: {}
      terminationGracePeriodSeconds: 30
status:
  availableReplicas: 2
:wq
```



```
student@node-1:~$ kubectl config use-context k8s
Switched to context "k8s".
student@node-1:~$ kubectl get deployments.apps
NAME           READY   UP-TO-DATE   AVAILABLE   AGE
front-end      2/2     2            2           5h57m
presentation   2/2     2            2           5h56m
student@node-1:~$ kubectl edit deployments.apps front-end
deployment.apps/front-end edited
student@node-1:~$ kubectl expose deployment front-end --name=front-end-svc --port=80 --type=NodePort --protocol=TCP
service/front-end-svc exposed
student@node-1:~$ kubectl describe svc front-end-svc
Name:                     front-end-svc
Namespace:                default
Labels:                   <none>
Annotations:              <none>
Selector:                 app=front-end
Type:                     NodePort
IP Family Policy:         SingleStack
IP Families:              IPv4
IP:                       10.107.66.230
```

Question 7 (Topic 1)

SIMULATION -

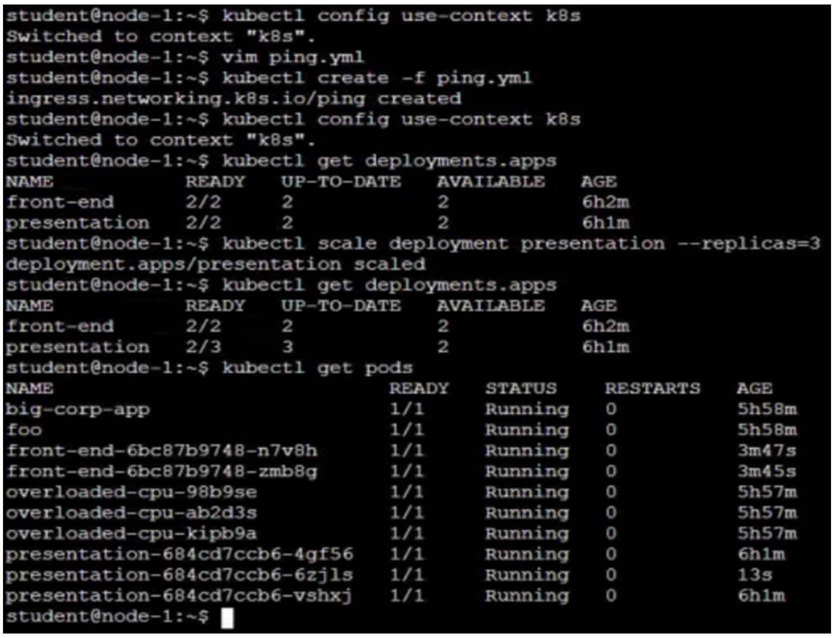
Set configuration context:

```
[student@node-1] $ | kubectl config use-context k8s
```

Task -
Scale the deployment presentation to 3 pods.

Answer : **See explanation below.**

Explanation:



```
student@node-1:~$ kubectl config use-context k8s
Switched to context "k8s".
student@node-1:~$ vim ping.yml
student@node-1:~$ kubectl create -f ping.yml
ingress.networking.k8s.io/ping created
student@node-1:~$ kubectl config use-context k8s
Switched to context "k8s".
student@node-1:~$ kubectl get deployments.apps
NAME           READY   UP-TO-DATE   AVAILABLE   AGE
front-end      2/2     2            2           6h2m
presentation   2/2     2            2           6h1m
student@node-1:~$ kubectl scale deployment presentation --replicas=3
deployment.apps/presentation scaled
student@node-1:~$ kubectl get deployments.apps
NAME           READY   UP-TO-DATE   AVAILABLE   AGE
front-end      2/2     2            2           6h2m
presentation   2/3     3            2           6h1m
student@node-1:~$ kubectl get pods
NAME                           READY   STATUS    RESTARTS   AGE
big-corp-app                   1/1     Running   0          5h58m
foo                            1/1     Running   0          5h58m
front-end-6bc87b9748-n7v8h     1/1     Running   0          3m47s
front-end-6bc87b9748-zmb8g     1/1     Running   0          3m45s
overloaded-cpu-98b9se          1/1     Running   0          5h57m
overloaded-cpu-ab2d3s          1/1     Running   0          5h57m
overloaded-cpu-kipb9a          1/1     Running   0          5h57m
presentation-684cd7ccb6-4gf56  1/1     Running   0          6h1m
presentation-684cd7ccb6-6zjls  1/1     Running   0          13s
presentation-684cd7ccb6-vshxj  1/1     Running   0          6h1m
student@node-1:~$ █
```

Question 8 (Topic 1)

SIMULATION -

Set configuration context:

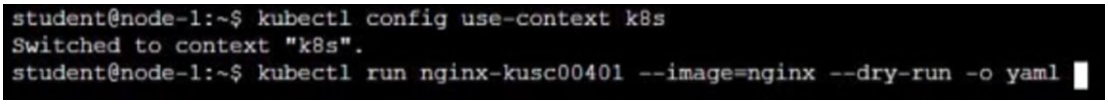
```
[student@node-1] $ | kubectl config use-context k8s
```

Task -
Schedule a pod as follows:
⇨ Name: nginx-kusc00401

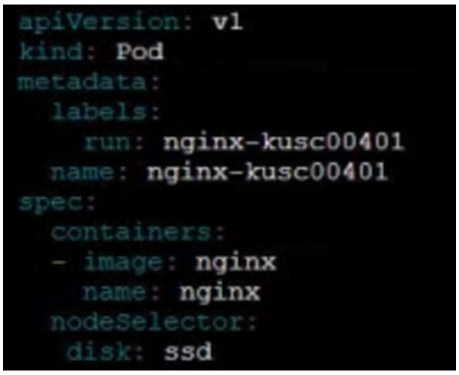
⇨ Image: nginx
⇨ Node selector: disk=ssd

Answer : **See explanation below.**

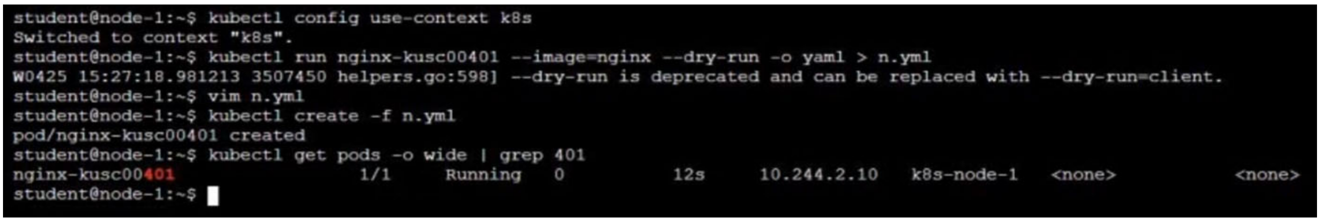
Explanation:



```
student@node-1:~$ kubectl config use-context k8s
Switched to context "k8s".
student@node-1:~$ kubectl run nginx-kusc00401 --image=nginx --dry-run -o yaml
```



```
apiVersion: v1
kind: Pod
metadata:
  labels:
    run: nginx-kusc00401
  name: nginx-kusc00401
spec:
  containers:
  - image: nginx
    name: nginx
  nodeSelector:
   disk: ssd
```



```
student@node-1:~$ kubectl config use-context k8s
Switched to context "k8s".
student@node-1:~$ kubectl run nginx-kusc00401 --image=nginx --dry-run -o yaml > n.yml
W0425 15:27:18.981213 3507450 helpers.go:598] --dry-run is deprecated and can be replaced with --dry-run=client.
student@node-1:~$ vim n.yml
student@node-1:~$ kubectl create -f n.yml
pod/nginx-kusc00401 created
student@node-1:~$ kubectl get pods -o wide | grep 401
nginx-kusc00401                1/1     Running   0          12s     10.244.2.10    k8s-node-1    <none>           <none>
student@node-1:~$
```

Question 9 (Topic 1)

SIMULATION -

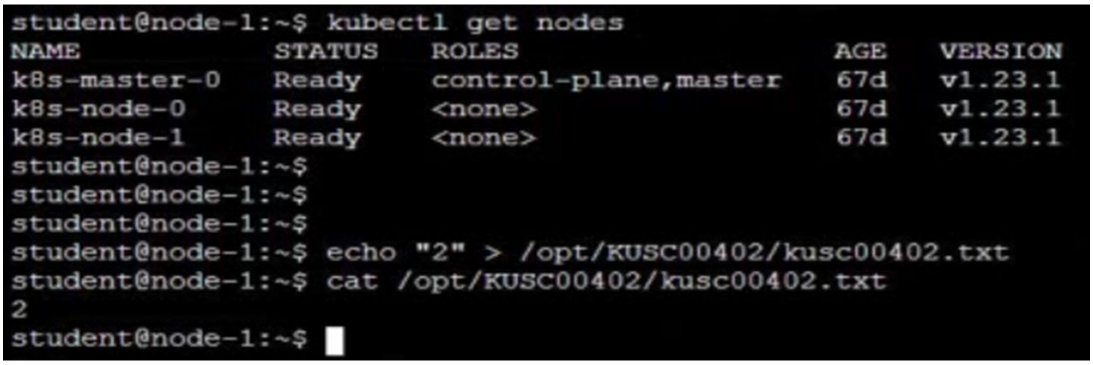
Set configuration context:

```
[student@node-1] $ | kubectl config u
se-context k8s
```

Task -
Check to see how many nodes are ready (not including nodes tainted NoSchedule) and write the number to /opt/KUSC00402/kusc00402.txt.

Answer : **See explanation below.**

Explanation:



```
student@node-1:~$ kubectl get nodes
NAME            STATUS    ROLES                     AGE    VERSION
k8s-master-0    Ready     control-plane,master      67d    v1.23.1
k8s-node-0      Ready     <none>                    67d    v1.23.1
k8s-node-1      Ready     <none>                    67d    v1.23.1
student@node-1:~$
student@node-1:~$
student@node-1:~$
student@node-1:~$ echo "2" > /opt/KUSC00402/kusc00402.txt
student@node-1:~$ cat /opt/KUSC00402/kusc00402.txt
2
student@node-1:~$
```

Question 10 (Topic 1)

SIMULATION -

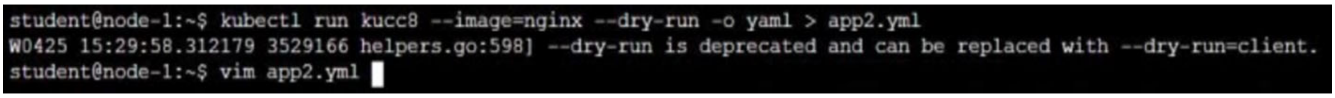
Set configuration context:

```
[student@node-1] $ | kubectl config use-context k8s
```

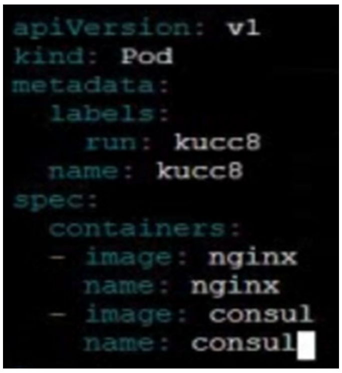
Task -
Schedule a Pod as follows:
⇛ Name: kucc8
⇛ App Containers: 2
⇛ Container Name/Images:
- nginx
- consul

Answer : **See explanation below.**

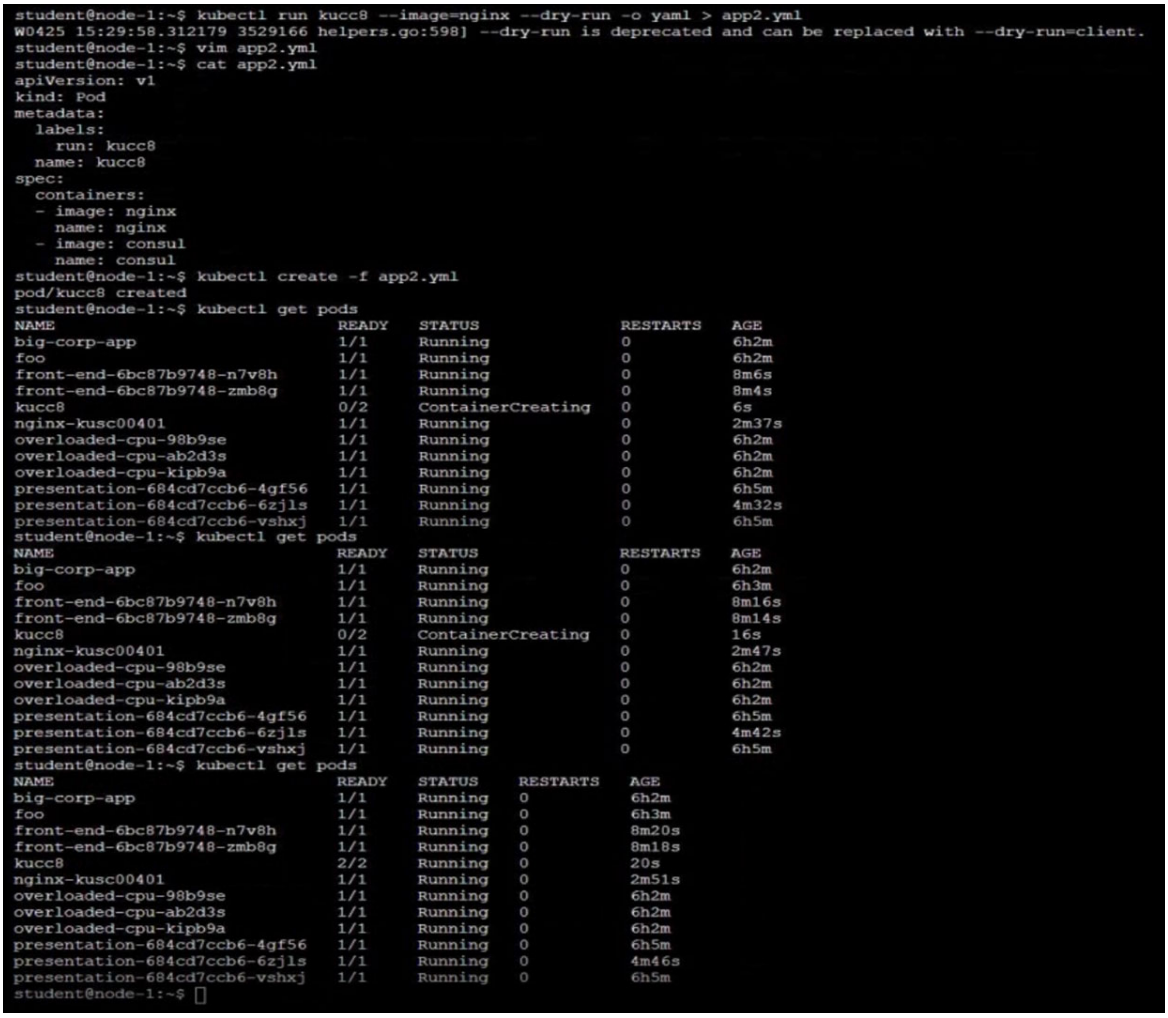
Explanation:



```
student@node-1:~$ kubectl run kucc8 --image=nginx --dry-run -o yaml > app2.yml
W0425 15:29:58.312179 3529166 helpers.go:598] --dry-run is deprecated and can be replaced with --dry-run=client.
student@node-1:~$ vim app2.yml
```



```
apiVersion: v1
kind: Pod
metadata:
  labels:
    run: kucc8
  name: kucc8
spec:
  containers:
  - image: nginx
    name: nginx
  - image: consul
    name: consul
```



```
student@node-1:~$ kubectl run kucc8 --image=nginx --dry-run -o yaml > app2.yml
W0425 15:29:58.312179 3529166 helpers.go:598] --dry-run is deprecated and can be replaced with --dry-run=client.
student@node-1:~$ vim app2.yml
student@node-1:~$ cat app2.yml
apiVersion: v1
kind: Pod
metadata:
  labels:
    run: kucc8
  name: kucc8
spec:
  containers:
  - image: nginx
    name: nginx
  - image: consul
    name: consul
student@node-1:~$ kubectl create -f app2.yml
pod/kucc8 created
student@node-1:~$ kubectl get pods
NAME                                READY   STATUS              RESTARTS   AGE
big-corp-app                        1/1     Running             0          6h2m
foo                                 1/1     Running             0          6h2m
front-end-6bc87b9748-n7v8h          1/1     Running             0          8m6s
front-end-6bc87b9748-zmb8g          1/1     Running             0          8m4s
kucc8                               0/2     ContainerCreating   0          6s
nginx-kusc00401                     1/1     Running             0          2m37s
overloaded-cpu-98b9se               1/1     Running             0          6h2m
overloaded-cpu-ab2d3s               1/1     Running             0          6h2m
overloaded-cpu-kipb9a               1/1     Running             0          6h2m
presentation-684cd7ccb6-4gf56       1/1     Running             0          6h5m
presentation-684cd7ccb6-6zjls       1/1     Running             0          4m32s
presentation-684cd7ccb6-vshxj       1/1     Running             0          6h5m
student@node-1:~$ kubectl get pods
NAME                                READY   STATUS              RESTARTS   AGE
big-corp-app                        1/1     Running             0          6h2m
foo                                 1/1     Running             0          6h3m
front-end-6bc87b9748-n7v8h          1/1     Running             0          8m16s
front-end-6bc87b9748-zmb8g          1/1     Running             0          8m14s
kucc8                               0/2     ContainerCreating   0          16s
nginx-kusc00401                     1/1     Running             0          2m47s
overloaded-cpu-98b9se               1/1     Running             0          6h2m
overloaded-cpu-ab2d3s               1/1     Running             0          6h2m
overloaded-cpu-kipb9a               1/1     Running             0          6h2m
presentation-684cd7ccb6-4gf56       1/1     Running             0          6h5m
presentation-684cd7ccb6-6zjls       1/1     Running             0          4m42s
presentation-684cd7ccb6-vshxj       1/1     Running             0          6h5m
student@node-1:~$ kubectl get pods
NAME                                READY   STATUS    RESTARTS   AGE
big-corp-app                        1/1     Running   0          6h2m
foo                                 1/1     Running   0          6h3m
front-end-6bc87b9748-n7v8h          1/1     Running   0          8m20s
front-end-6bc87b9748-zmb8g          1/1     Running   0          8m18s
kucc8                               2/2     Running   0          20s
nginx-kusc00401                     1/1     Running   0          2m51s
overloaded-cpu-98b9se               1/1     Running   0          6h2m
overloaded-cpu-ab2d3s               1/1     Running   0          6h2m
overloaded-cpu-kipb9a               1/1     Running   0          6h2m
presentation-684cd7ccb6-4gf56       1/1     Running   0          6h5m
presentation-684cd7ccb6-6zjls       1/1     Running   0          4m46s
presentation-684cd7ccb6-vshxj       1/1     Running   0          6h5m
student@node-1:~$ 
```

Page: 2 / 4
Total 20 questions

Previous Page Next Page

10 questions per page

Certified Kubernetes Administrator v1.0 (CKA)

Page: 3 / 4
Total 20 questions

10 questions per page

Question 11 (Topic 1)

SIMULATION -

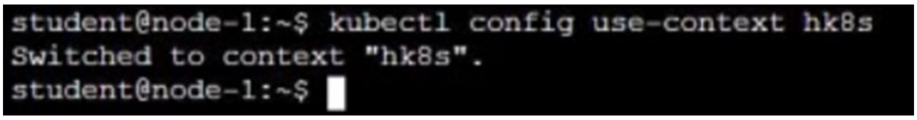
Set configuration context:

```
[student@node-1] $ | kubectl config use-context hk8s
```

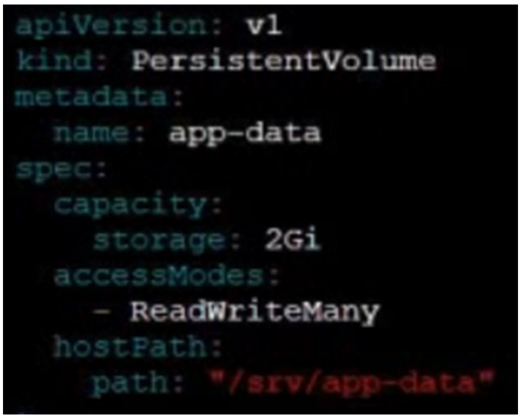
Task -
Create a persistent volume with name app-data, of capacity 2Gi and access mode ReadOnlyMany. The type of volume is hostPath and its location is /srv/app- data.

Answer : **See explanation below.**

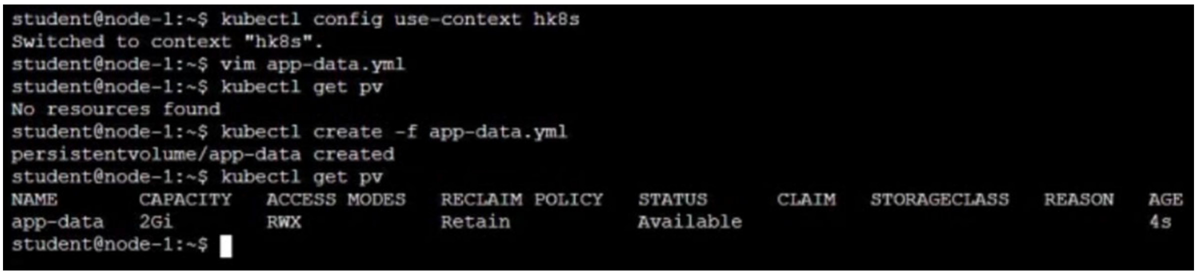
Explanation:



```
student@node-1:~$ kubectl config use-context hk8s
Switched to context "hk8s".
student@node-1:~$
```



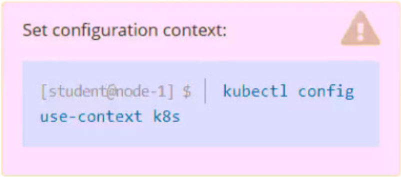
```
apiVersion: v1
kind: PersistentVolume
metadata:
  name: app-data
spec:
  capacity:
    storage: 2Gi
  accessModes:
    - ReadWriteMany
  hostPath:
    path: "/srv/app-data"
```



```
student@node-1:~$ kubectl config use-context hk8s
Switched to context "hk8s".
student@node-1:~$ vim app-data.yml
student@node-1:~$ kubectl get pv
No resources found
student@node-1:~$ kubectl create -f app-data.yml
persistentvolume/app-data created
student@node-1:~$ kubectl get pv
NAME       CAPACITY   ACCESS MODES   RECLAIM POLICY   STATUS      CLAIM   STORAGECLASS   REASON   AGE
app-data   2Gi        RWX            Retain           Available                                   4s
student@node-1:~$
```

Question 12 (Topic 1)

SIMULATION -



Set configuration context:

```
[student@node-1] $ | kubectl config
use-context k8s
```

Task -

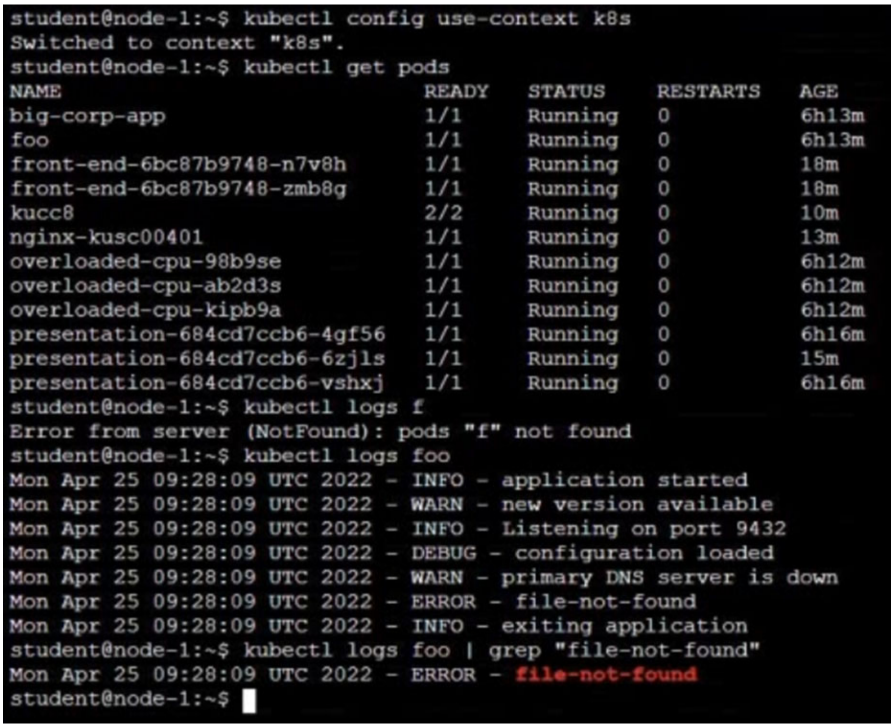
Monitor the logs of pod foo and:

⇨ Extract log lines corresponding to error file-not-found

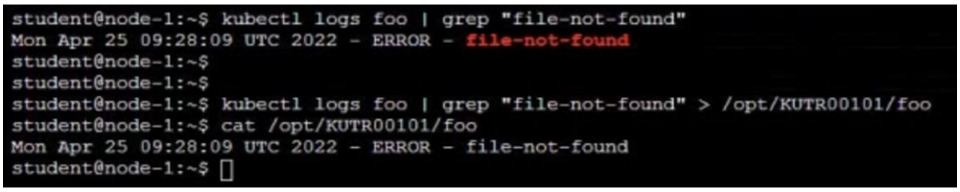
⇨ Write them to /opt/KUTR00101/foo

Answer : **See explanation below.**

Explanation:



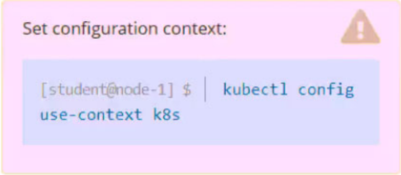
```
student@node-1:~$ kubectl config use-context k8s
Switched to context "k8s".
student@node-1:~$ kubectl get pods
NAME                             READY   STATUS    RESTARTS   AGE
big-corp-app                     1/1     Running   0          6h13m
foo                              1/1     Running   0          6h13m
front-end-6bc87b9748-n7v8h       1/1     Running   0          18m
front-end-6bc87b9748-zmb8g       1/1     Running   0          18m
kucc8                            2/2     Running   0          10m
nginx-kusc00401                  1/1     Running   0          13m
overloaded-cpu-98b9se            1/1     Running   0          6h12m
overloaded-cpu-ab2d3s            1/1     Running   0          6h12m
overloaded-cpu-kipb9a            1/1     Running   0          6h12m
presentation-684cd7ccb6-4gf56    1/1     Running   0          6h16m
presentation-684cd7ccb6-6zjls    1/1     Running   0          15m
presentation-684cd7ccb6-vshxj    1/1     Running   0          6h16m
student@node-1:~$ kubectl logs f
Error from server (NotFound): pods "f" not found
student@node-1:~$ kubectl logs foo
Mon Apr 25 09:28:09 UTC 2022 - INFO - application started
Mon Apr 25 09:28:09 UTC 2022 - WARN - new version available
Mon Apr 25 09:28:09 UTC 2022 - INFO - Listening on port 9432
Mon Apr 25 09:28:09 UTC 2022 - DEBUG - configuration loaded
Mon Apr 25 09:28:09 UTC 2022 - WARN - primary DNS server is down
Mon Apr 25 09:28:09 UTC 2022 - ERROR - file-not-found
Mon Apr 25 09:28:09 UTC 2022 - INFO - exiting application
student@node-1:~$ kubectl logs foo | grep "file-not-found"
Mon Apr 25 09:28:09 UTC 2022 - ERROR - file-not-found
student@node-1:~$
```



```
student@node-1:~$ kubectl logs foo | grep "file-not-found"
Mon Apr 25 09:28:09 UTC 2022 - ERROR - file-not-found
student@node-1:~$
student@node-1:~$
student@node-1:~$ kubectl logs foo | grep "file-not-found" > /opt/KUTR00101/foo
student@node-1:~$ cat /opt/KUTR00101/foo
Mon Apr 25 09:28:09 UTC 2022 - ERROR - file-not-found
student@node-1:~$
```

Question 13 (Topic 1)

SIMULATION -



Set configuration context:

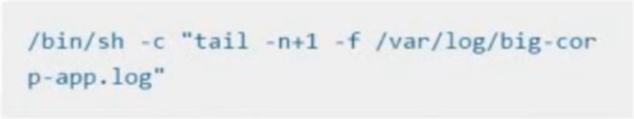
```
[student@node-1] $ | kubectl config
use-context k8s
```

Context -

An existing Pod needs to be integrated into the Kubernetes built-in logging architecture (e.g. kubectl logs). Adding a streaming sidecar container is a good and common way to accomplish this requirement.

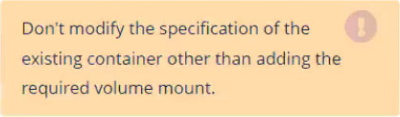
Task -

Add a sidecar container named sidecar, using the busybox image, to the existing Pod big-corp-app. The new sidecar container has to run the following command:



```
/bin/sh -c "tail -n+1 -f /var/log/big-cor
p-app.log"
```

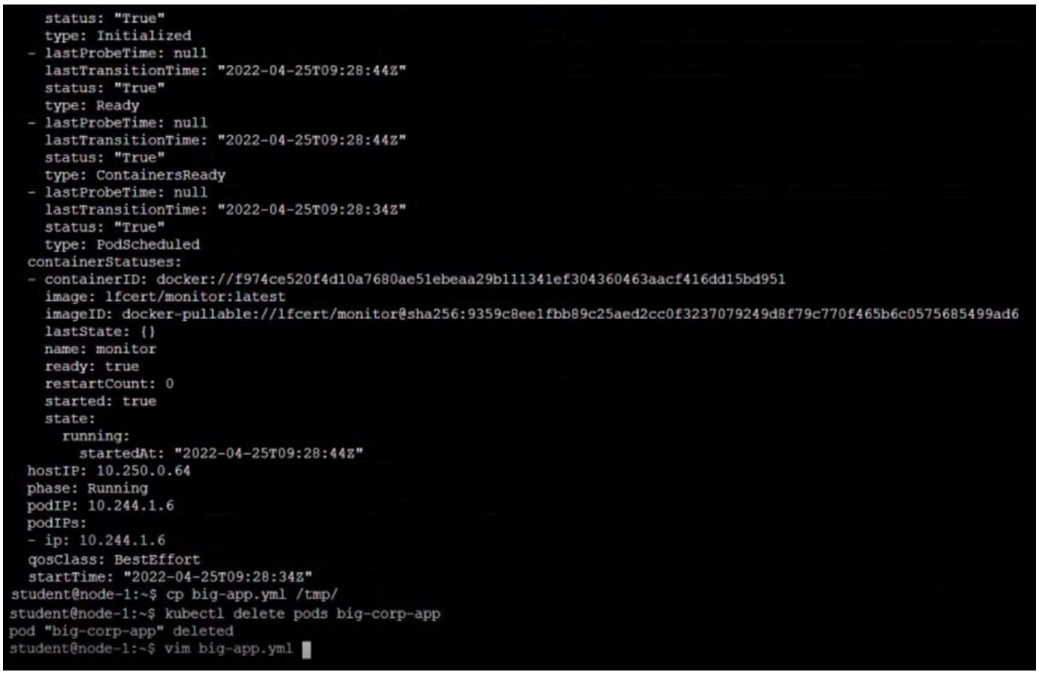
Use a Volume, mounted at /var/log, to make the log file big-corp-app.log available to the sidecar container.



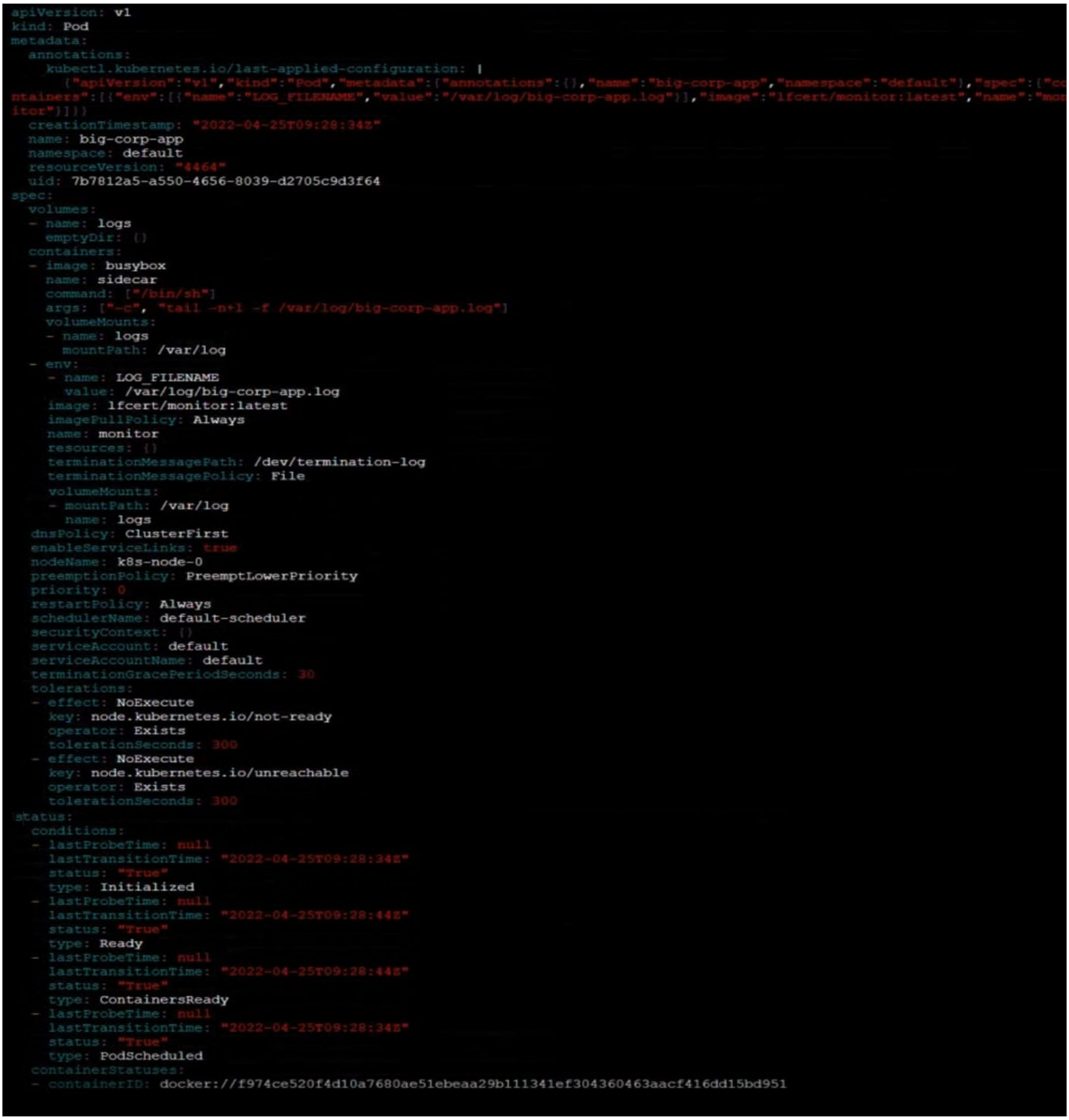
Don't modify the specification of the existing container other than adding the required volume mount.

Answer : **See explanation below.**

Explanation:



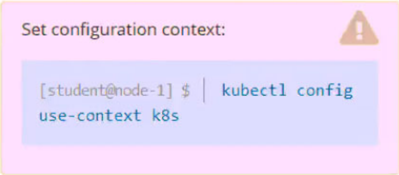
```
    status: "True"
    type: Initialized
  - lastProbeTime: null
    lastTransitionTime: "2022-04-25T09:28:44Z"
    status: "True"
    type: Ready
  - lastProbeTime: null
    lastTransitionTime: "2022-04-25T09:28:44Z"
    status: "True"
    type: ContainersReady
  - lastProbeTime: null
    lastTransitionTime: "2022-04-25T09:28:34Z"
    status: "True"
    type: PodScheduled
  containerStatuses:
  - containerID: docker://f974ce520f4d10a7680ae51ebeaa29b111341ef304360463aacf416dd15bd951
    image: lfcert/monitor:latest
    imageID: docker-pullable://lfcert/monitor@sha256:9359c8ee1fbb89c25aed2cc0f3237079249d8f79c770f465b6c0575685499ad6
    lastState: {}
    name: monitor
    ready: true
    restartCount: 0
    started: true
    state:
      running:
        startedAt: "2022-04-25T09:28:44Z"
  hostIP: 10.250.0.64
  phase: Running
  podIP: 10.244.1.6
  podIPs:
  - ip: 10.244.1.6
  qosClass: BestEffort
  startTime: "2022-04-25T09:28:34Z"
student@node-1:~$ cp big-app.yml /tmp/
student@node-1:~$ kubectl delete pods big-corp-app
pod "big-corp-app" deleted
student@node-1:~$ vim big-app.yml
```



```
apiVersion: v1
kind: Pod
metadata:
  annotations:
    kubectl.kubernetes.io/last-applied-configuration: |
      {"apiVersion":"v1","kind":"Pod","metadata":{"annotations":{},"name":"big-corp-app","namespace":"default"},"spec":{"co
ntainers":[{"env":[{"name":"LOG_FILENAME","value":"/var/log/big-corp-app.log"}],"image":"lfcert/monitor:latest","name":"mon
itor"}]}}
  creationTimestamp: "2022-04-25T09:28:34Z"
  name: big-corp-app
  namespace: default
  resourceVersion: "4464"
  uid: 7b7812a5-a550-4656-8039-d2705c9d3f64
spec:
  volumes:
  - name: logs
    emptyDir: {}
  containers:
  - image: busybox
    name: sidecar
    command: ["/bin/sh"]
    args: ["-c", "tail -n+1 -f /var/log/big-corp-app.log"]
    volumeMounts:
    - name: logs
      mountPath: /var/log
  - env:
    - name: LOG_FILENAME
      value: /var/log/big-corp-app.log
    image: lfcert/monitor:latest
    imagePullPolicy: Always
    name: monitor
    resources: {}
    terminationMessagePath: /dev/termination-log
    terminationMessagePolicy: File
    volumeMounts:
    - mountPath: /var/log
      name: logs
  dnsPolicy: ClusterFirst
  enableServiceLinks: true
  nodeName: k8s-node-0
  preemptionPolicy: PreemptLowerPriority
  priority: 0
  restartPolicy: Always
  schedulerName: default-scheduler
  securityContext: {}
  serviceAccount: default
  serviceAccountName: default
  terminationGracePeriodSeconds: 30
  tolerations:
  - effect: NoExecute
    key: node.kubernetes.io/not-ready
    operator: Exists
    tolerationSeconds: 300
  - effect: NoExecute
    key: node.kubernetes.io/unreachable
    operator: Exists
    tolerationSeconds: 300
status:
  conditions:
  - lastProbeTime: null
    lastTransitionTime: "2022-04-25T09:28:34Z"
    status: "True"
    type: Initialized
  - lastProbeTime: null
    lastTransitionTime: "2022-04-25T09:28:44Z"
    status: "True"
    type: Ready
  - lastProbeTime: null
    lastTransitionTime: "2022-04-25T09:28:44Z"
    status: "True"
    type: ContainersReady
  - lastProbeTime: null
    lastTransitionTime: "2022-04-25T09:28:34Z"
    status: "True"
    type: PodScheduled
  containerStatuses:
  - containerID: docker://f974ce520f4d10a7680ae51ebeaa29b111341ef304360463aacf416dd15bd951
```

Question 14 (Topic 1)

SIMULATION -



Set configuration context:

```
[student@node-1] $ | kubectl config
use-context k8s
```

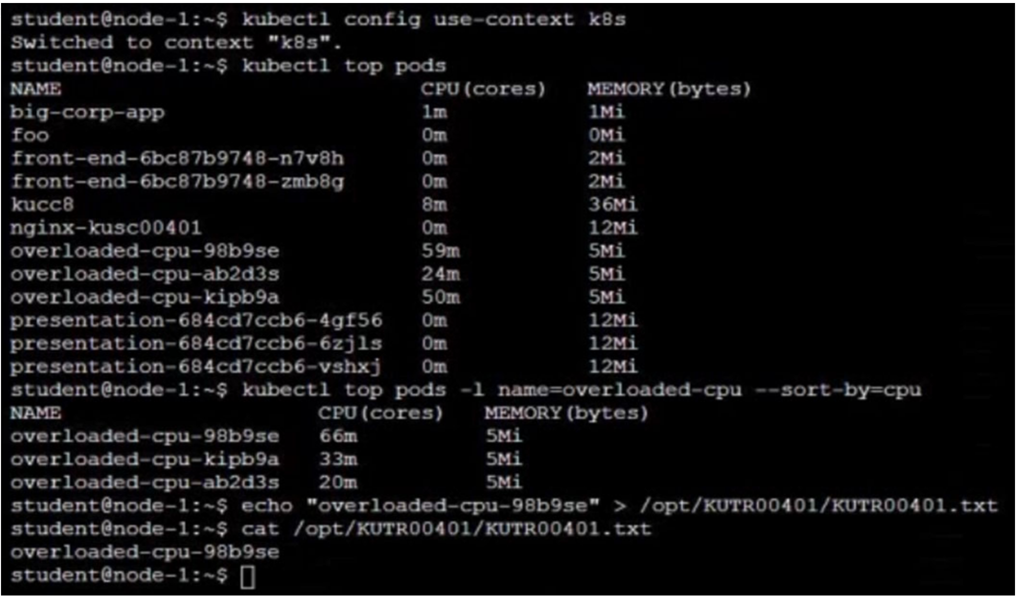
Answer : **See explanation below.**

Task -

Explanation:

From the pod label name=overloaded-cpu, find pods running high CPU workloads and write the name of the pod consuming most CPU to the file /opt/

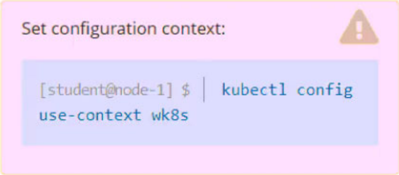
K



```
student@node-1:~$ kubectl config use-context k8s
Switched to context "k8s".
student@node-1:~$ kubectl top pods
NAME                              CPU(cores)    MEMORY(bytes)
big-corp-app                      1m            1Mi
foo                               0m            0Mi
front-end-6bc87b9748-n7v8h        0m            2Mi
front-end-6bc87b9748-zmb8g        0m            2Mi
kucc8                             8m            36Mi
nginx-kusc00401                   0m            12Mi
overloaded-cpu-98b9se             59m           5Mi
overloaded-cpu-ab2d3s             24m           5Mi
overloaded-cpu-kipb9a             50m           5Mi
presentation-684cd7ccb6-4gf56     0m            12Mi
presentation-684cd7ccb6-6zjls     0m            12Mi
presentation-684cd7ccb6-vshxj     0m            12Mi
student@node-1:~$ kubectl top pods -l name=overloaded-cpu --sort-by=cpu
NAME                      CPU(cores)    MEMORY(bytes)
overloaded-cpu-98b9se     66m           5Mi
overloaded-cpu-kipb9a     33m           5Mi
overloaded-cpu-ab2d3s     20m           5Mi
student@node-1:~$ echo "overloaded-cpu-98b9se" > /opt/KUTR00401/KUTR00401.txt
student@node-1:~$ cat /opt/KUTR00401/KUTR00401.txt
overloaded-cpu-98b9se
student@node-1:~$ 
```

Question 15 (Topic 1)

SIMULATION -



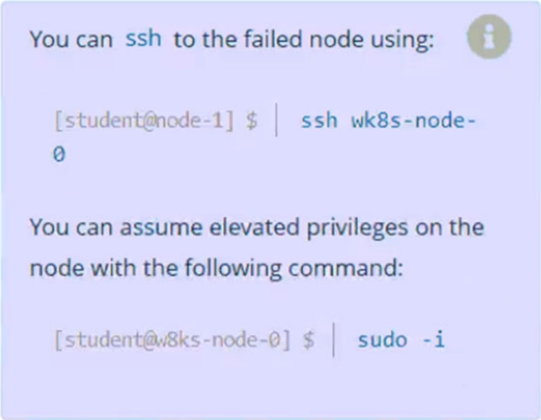
Set configuration context:

```
[student@node-1] $ | kubectl config
use-context wk8s
```

Task -

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:

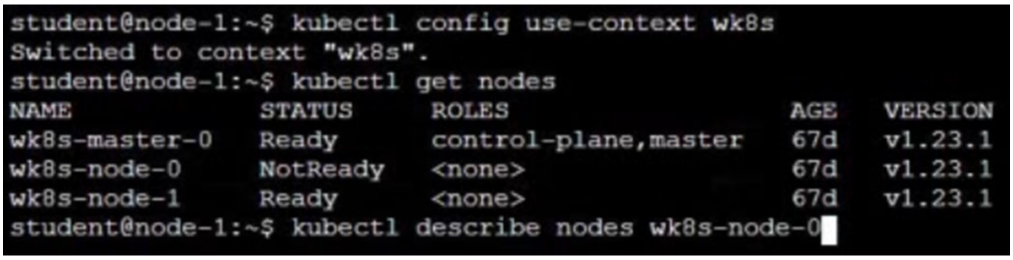
```
[student@node-1] $ | ssh wk8s-node-
0
```

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

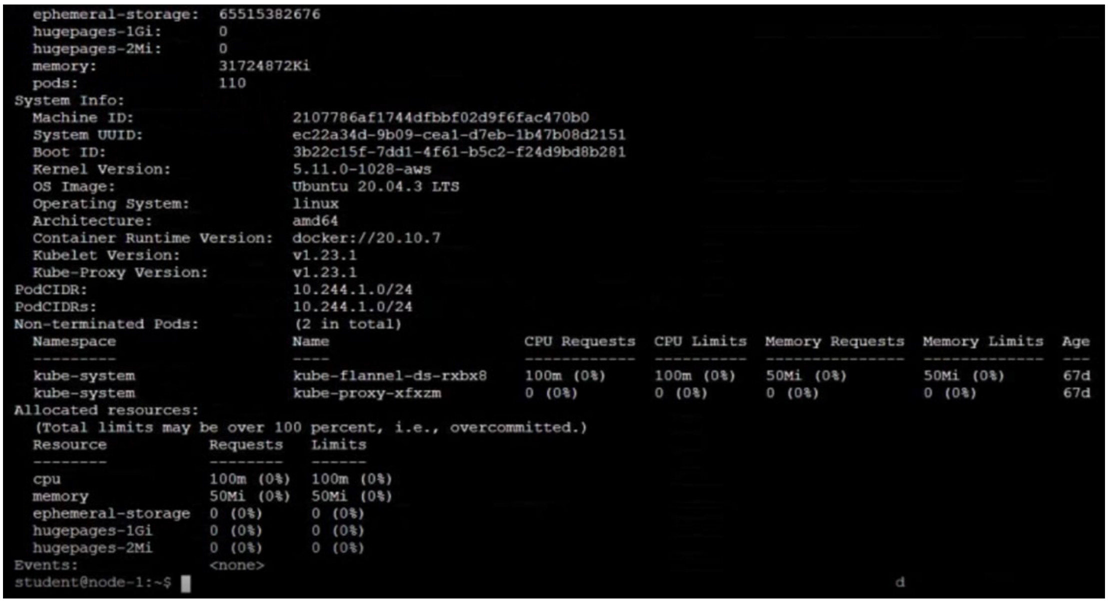
```
[student@w8ks-node-0] $ | sudo -i
```

Answer : **See explanation below.**

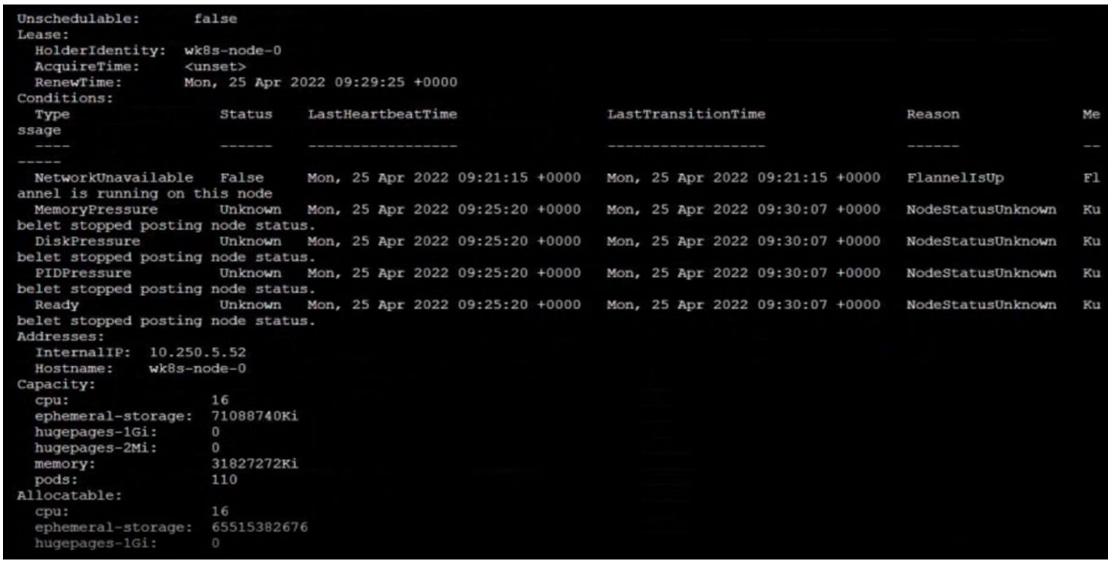
Explanation:



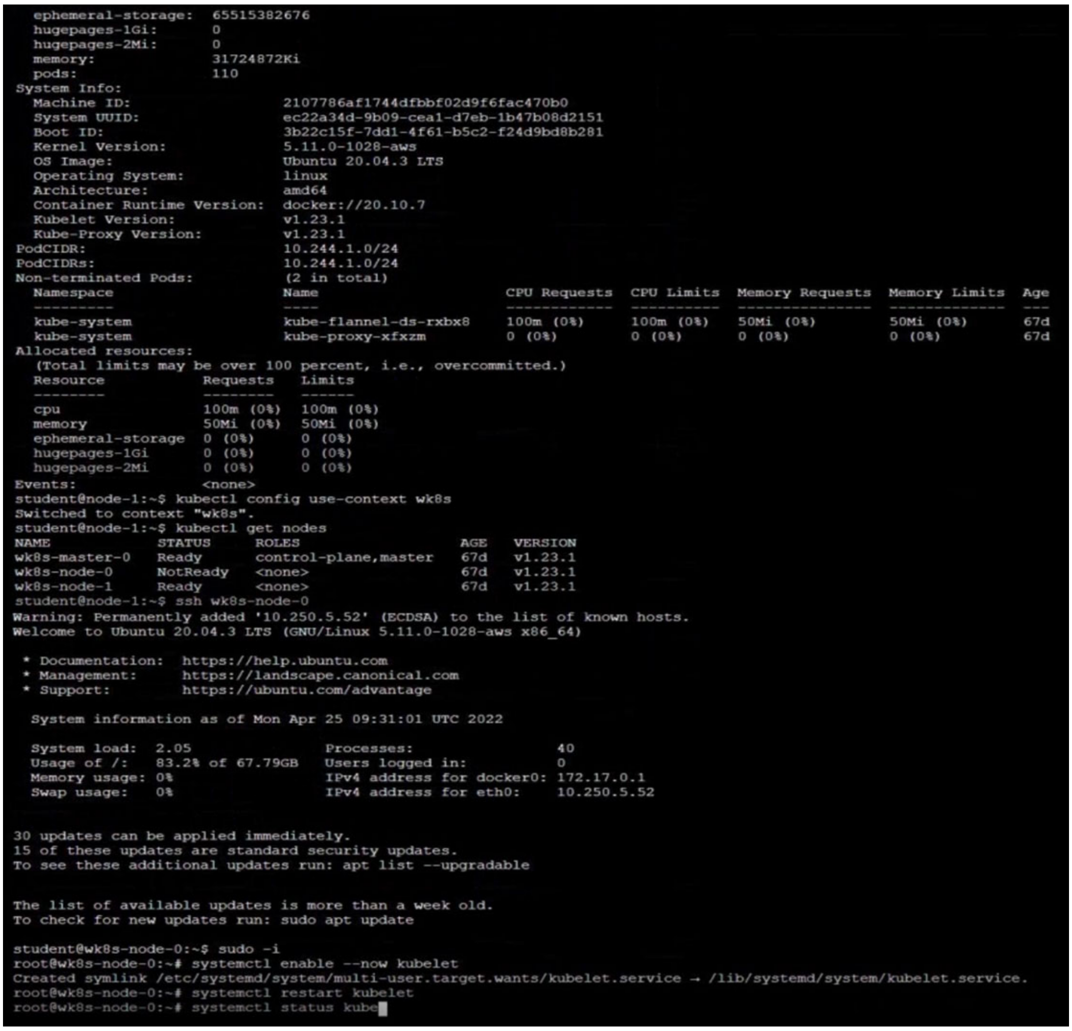
```
student@node-1:~$ kubectl config use-context wk8s
Switched to context "wk8s".
student@node-1:~$ kubectl get nodes
NAME            STATUS     ROLES                  AGE   VERSION
wk8s-master-0   Ready      control-plane,master   67d   v1.23.1
wk8s-node-0     NotReady   <none>                 67d   v1.23.1
wk8s-node-1     Ready      <none>                 67d   v1.23.1
student@node-1:~$ kubectl describe nodes wk8s-node-0
```



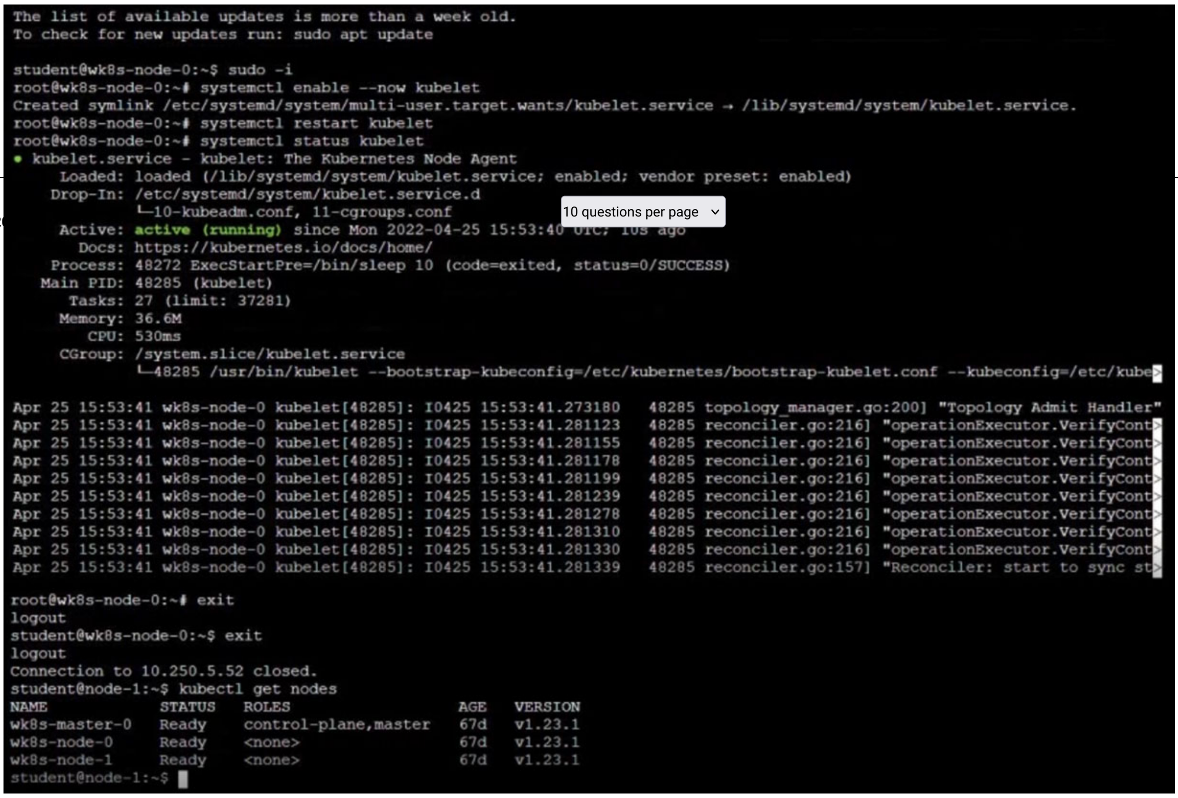
```
  ephemeral-storage:  65515382676
  hugepages-1Gi:      0
  hugepages-2Mi:      0
  memory:             31724872Ki
  pods:               110
System Info:
  Machine ID:                 2107786af1744dfbbf02d9f6fac470b0
  System UUID:                ec22a34d-9b09-cea1-d7eb-1b47b08d2151
  Boot ID:                    3b22c15f-7dd1-4f61-b5c2-f24d9bd8b281
  Kernel Version:             5.11.0-1028-aws
  OS Image:                   Ubuntu 20.04.3 LTS
  Operating System:           linux
  Architecture:               amd64
  Container Runtime Version:  docker://20.10.7
  Kubelet Version:            v1.23.1
  Kube-Proxy Version:         v1.23.1
PodCIDR:                      10.244.1.0/24
PodCIDRs:                     10.244.1.0/24
Non-terminated Pods:          (2 in total)
  Namespace                   Name                     CPU Requests  CPU Limits  Memory Requests  Memory Limits  Age
  ---------                   ----                     ------------  ----------  ---------------  -------------  ---
  kube-system                 kube-flannel-ds-rxbx8    100m (0%)     100m (0%)   50Mi (0%)        50Mi (0%)      67d
  kube-system                 kube-proxy-xfxzm         0 (0%)        0 (0%)      0 (0%)           0 (0%)         67d
Allocated resources:
  (Total limits may be over 100 percent, i.e., overcommitted.)
  Resource           Requests   Limits
  --------           --------   ------
  cpu                100m (0%)  100m (0%)
  memory             50Mi (0%)  50Mi (0%)
  ephemeral-storage  0 (0%)     0 (0%)
  hugepages-1Gi      0 (0%)     0 (0%)
  hugepages-2Mi      0 (0%)     0 (0%)
Events:              <none>
student@node-1:~$                                                                                  d
```



```
Unschedulable:      false
Lease:
  HolderIdentity:  wk8s-node-0
  AcquireTime:     <unset>
  RenewTime:       Mon, 25 Apr 2022 09:29:25 +0000
Conditions:
  Type                 Status    LastHeartbeatTime                  LastTransitionTime                 Reason              Me
ssage
  ----                 ------    -----------------                  ------------------                 ------              --
-----
  NetworkUnavailable   False     Mon, 25 Apr 2022 09:21:15 +0000    Mon, 25 Apr 2022 09:21:15 +0000    FlannelIsUp         Fl
annel is running on this node
  MemoryPressure       Unknown   Mon, 25 Apr 2022 09:25:20 +0000    Mon, 25 Apr 2022 09:30:07 +0000    NodeStatusUnknown   Ku
belet stopped posting node status.
  DiskPressure         Unknown   Mon, 25 Apr 2022 09:25:20 +0000    Mon, 25 Apr 2022 09:30:07 +0000    NodeStatusUnknown   Ku
belet stopped posting node status.
  PIDPressure          Unknown   Mon, 25 Apr 2022 09:25:20 +0000    Mon, 25 Apr 2022 09:30:07 +0000    NodeStatusUnknown   Ku
belet stopped posting node status.
  Ready                Unknown   Mon, 25 Apr 2022 09:25:20 +0000    Mon, 25 Apr 2022 09:30:07 +0000    NodeStatusUnknown   Ku
belet stopped posting node status.
Addresses:
  InternalIP:  10.250.5.52
  Hostname:    wk8s-node-0
Capacity:
  cpu:                16
  ephemeral-storage:  71088740Ki
  hugepages-1Gi:      0
  hugepages-2Mi:      0
  memory:             31827272Ki
  pods:               110
Allocatable:
  cpu:                16
  ephemeral-storage:  65515382676
  hugepages-1Gi:      0
```



```
  ephemeral-storage:  65515382676
  hugepages-1Gi:      0
  hugepages-2Mi:      0
  memory:             31724872Ki
  pods:               110
System Info:
  Machine ID:                 2107786af1744dfbbf02d9f6fac470b0
  System UUID:                ec22a34d-9b09-cea1-d7eb-1b47b08d2151
  Boot ID:                    3b22c15f-7dd1-4f61-b5c2-f24d9bd8b281
  Kernel Version:             5.11.0-1028-aws
  OS Image:                   Ubuntu 20.04.3 LTS
  Operating System:           linux
  Architecture:               amd64
  Container Runtime Version:  docker://20.10.7
  Kubelet Version:            v1.23.1
  Kube-Proxy Version:         v1.23.1
PodCIDR:                      10.244.1.0/24
PodCIDRs:                     10.244.1.0/24
Non-terminated Pods:          (2 in total)
  Namespace                   Name                         CPU Requests  CPU Limits  Memory Requests  Memory Limits  Age
  ---------                   ----                         ------------  ----------  ---------------  -------------  ---
  kube-system                 kube-flannel-ds-rxbx8        100m (0%)     100m (0%)   50Mi (0%)        50Mi (0%)      67d
  kube-system                 kube-proxy-xfxzm             0 (0%)        0 (0%)      0 (0%)           0 (0%)         67d
Allocated resources:
  (Total limits may be over 100 percent, i.e., overcommitted.)
  Resource           Requests   Limits
  --------           --------   ------
  cpu                100m (0%)  100m (0%)
  memory             50Mi (0%)  50Mi (0%)
  ephemeral-storage  0 (0%)     0 (0%)
  hugepages-1Gi      0 (0%)     0 (0%)
  hugepages-2Mi      0 (0%)     0 (0%)
Events:              <none>
student@node-1:~$ kubectl config use-context wk8s
Switched to context "wk8s".
student@node-1:~$ kubectl get nodes
NAME            STATUS     ROLES                  AGE   VERSION
wk8s-master-0   Ready      control-plane,master   67d   v1.23.1
wk8s-node-0     NotReady   <none>                 67d   v1.23.1
wk8s-node-1     Ready      <none>                 67d   v1.23.1
student@node-1:~$ ssh wk8s-node-0
Warning: Permanently added '10.250.5.52' (ECDSA) to the list of known hosts.
Welcome to Ubuntu 20.04.3 LTS (GNU/Linux 5.11.0-1028-aws x86_64)

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

  System information as of Mon Apr 25 09:31:01 UTC 2022

  System load:  2.05              Processes:                40
  Usage of /:   83.2% of 67.79GB  Users logged in:          0
  Memory usage: 0%                IPv4 address for docker0: 172.17.0.1
  Swap usage:   0%                IPv4 address for eth0:    10.250.5.52


30 updates can be applied immediately.
15 of these updates are standard security updates.
To see these additional updates run: apt list --upgradable


The list of available updates is more than a week old.
To check for new updates run: sudo apt update

student@wk8s-node-0:~$ sudo -i
root@wk8s-node-0:~# systemctl enable --now kubelet
Created symlink /etc/systemd/system/multi-user.target.wants/kubelet.service → /lib/systemd/system/kubelet.service.
root@wk8s-node-0:~# systemctl restart kubelet
root@wk8s-node-0:~# systemctl status kube
```



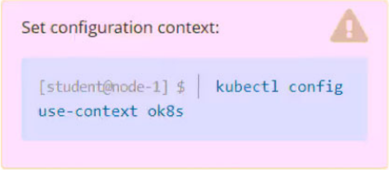
```
The list of available updates is more than a week old.
To check for new updates run: sudo apt update

student@wk8s-node-0:~$ sudo -i
root@wk8s-node-0:~# systemctl enable --now kubelet
Created symlink /etc/systemd/system/multi-user.target.wants/kubelet.service → /lib/systemd/system/kubelet.service.
root@wk8s-node-0:~# systemctl restart kubelet
root@wk8s-node-0:~# systemctl status kubelet
● kubelet.service - kubelet: The Kubernetes Node Agent
     Loaded: loaded (/lib/systemd/system/kubelet.service; enabled; vendor preset: enabled)
    Drop-In: /etc/systemd/system/kubelet.service.d
             └─10-kubeadm.conf, 11-cgroups.conf
     Active: active (running) since Mon 2022-04-25 15:53:40 UTC; 10s ago
       Docs: https://kubernetes.io/docs/home/
    Process: 48272 ExecStartPre=/bin/sleep 10 (code=exited, status=0/SUCCESS)
   Main PID: 48285 (kubelet)
      Tasks: 27 (limit: 37281)
     Memory: 36.6M
        CPU: 530ms
     CGroup: /system.slice/kubelet.service
             └─48285 /usr/bin/kubelet --bootstrap-kubeconfig=/etc/kubernetes/bootstrap-kubelet.conf --kubeconfig=/etc/kube>

Apr 25 15:53:41 wk8s-node-0 kubelet[48285]: I0425 15:53:41.273180   48285 topology_manager.go:200] "Topology Admit Handler"
Apr 25 15:53:41 wk8s-node-0 kubelet[48285]: I0425 15:53:41.281123   48285 reconciler.go:216] "operationExecutor.VerifyCont>
Apr 25 15:53:41 wk8s-node-0 kubelet[48285]: I0425 15:53:41.281155   48285 reconciler.go:216] "operationExecutor.VerifyCont>
Apr 25 15:53:41 wk8s-node-0 kubelet[48285]: I0425 15:53:41.281178   48285 reconciler.go:216] "operationExecutor.VerifyCont>
Apr 25 15:53:41 wk8s-node-0 kubelet[48285]: I0425 15:53:41.281199   48285 reconciler.go:216] "operationExecutor.VerifyCont>
Apr 25 15:53:41 wk8s-node-0 kubelet[48285]: I0425 15:53:41.281239   48285 reconciler.go:216] "operationExecutor.VerifyCont>
Apr 25 15:53:41 wk8s-node-0 kubelet[48285]: I0425 15:53:41.281278   48285 reconciler.go:216] "operationExecutor.VerifyCont>
Apr 25 15:53:41 wk8s-node-0 kubelet[48285]: I0425 15:53:41.281310   48285 reconciler.go:216] "operationExecutor.VerifyCont>
Apr 25 15:53:41 wk8s-node-0 kubelet[48285]: I0425 15:53:41.281330   48285 reconciler.go:216] "operationExecutor.VerifyCont>
Apr 25 15:53:41 wk8s-node-0 kubelet[48285]: I0425 15:53:41.281339   48285 reconciler.go:157] "Reconciler: start to sync st>

root@wk8s-node-0:~# exit
logout
student@wk8s-node-0:~$ exit
logout
Connection to 10.250.5.52 closed.
student@node-1:~$ kubectl get nodes
NAME            STATUS   ROLES                  AGE   VERSION
wk8s-master-0   Ready    control-plane,master   67d   v1.23.1
wk8s-node-0     Ready    <none>                 67d   v1.23.1
wk8s-node-1     Ready    <none>                 67d   v1.23.1
student@node-1:~$
```

Page:
Total 2
10 questions per page

Certified Kubernetes Administrator v1.0 (CKA)

Question 16 (Topic 1)

SIMULATION -



Set configuration context:

```
[student@node-1] $ kubectl config use-context ok8s
```

Task -

Create a new PersistentVolumeClaim:

⇨ Name: pv-volume
⇨ Class: csi-hostpath-sc
⇨ Capacity: 10Mi

Create a new Pod which mounts the PersistentVolumeClaim as a volume:

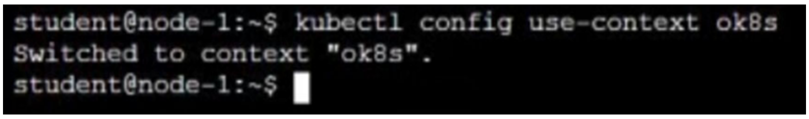
⇨ Name: web-server
⇨ Image: nginx
⇨ Mount path: /usr/share/nginx/html

Configure the new Pod to have ReadWriteOnce access on the volume.

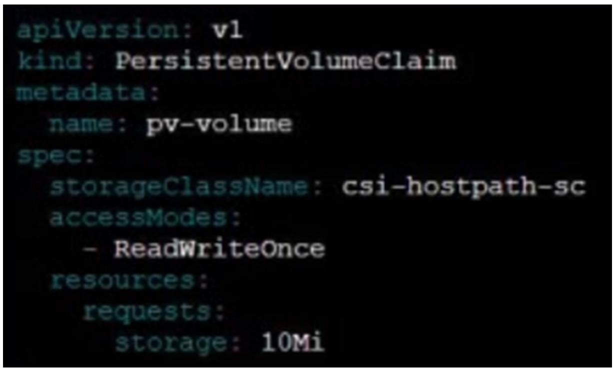
Finally, using kubectl edit or kubectl patch expand the PersistentVolumeClaim to a capacity of 70Mi and record that change.

Answer : **See explanation below.**

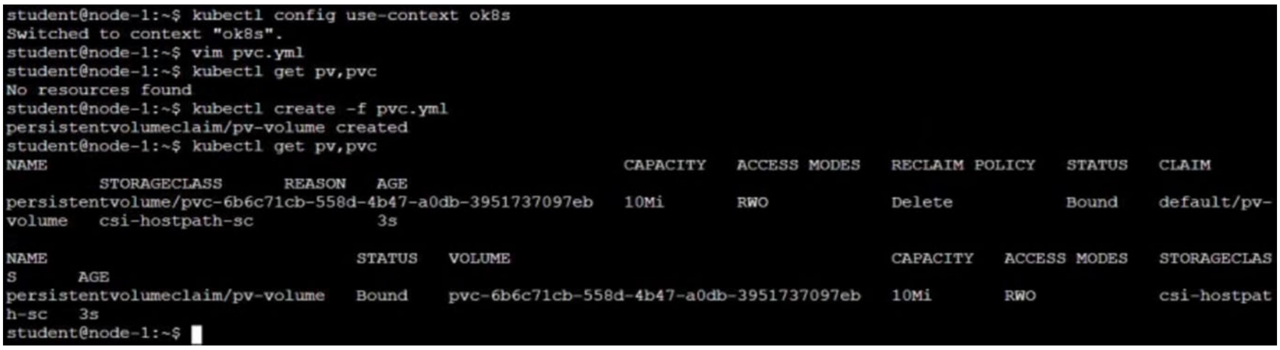
Explanation:



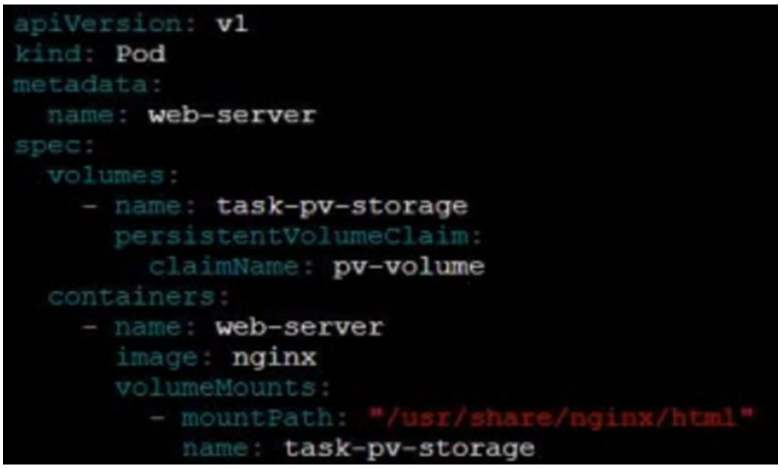
```
student@node-1:~$ kubectl config use-context ok8s
Switched to context "ok8s".
student@node-1:~$
```



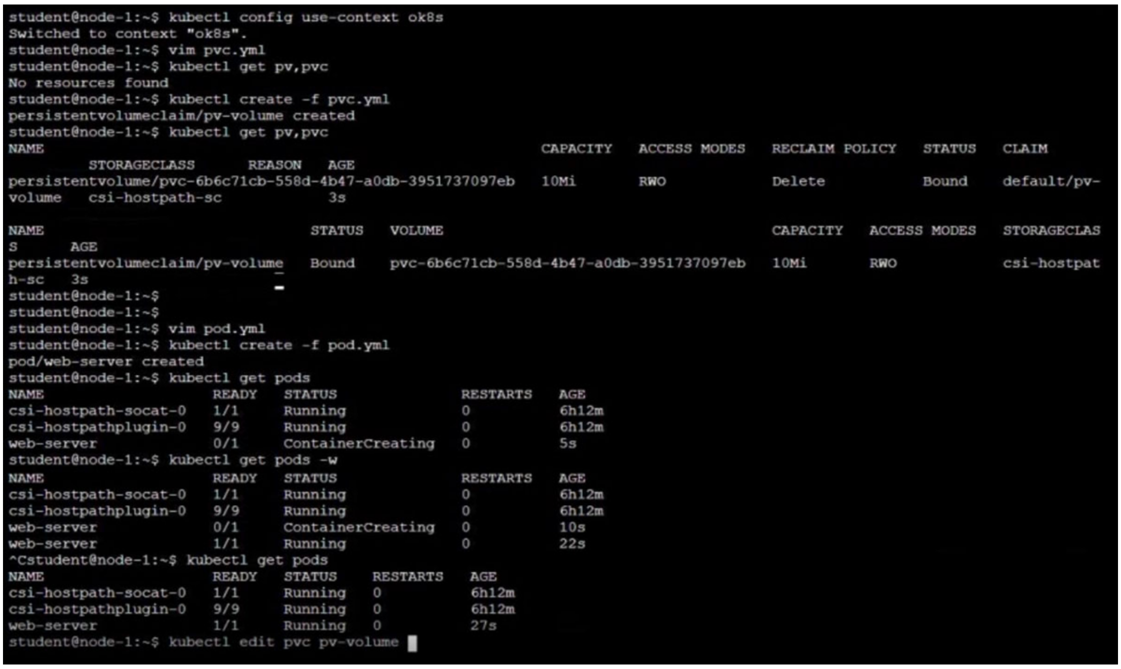
```
apiVersion: v1
kind: PersistentVolumeClaim
metadata:
  name: pv-volume
spec:
  storageClassName: csi-hostpath-sc
  accessModes:
    - ReadWriteOnce
  resources:
    requests:
      storage: 10Mi
```



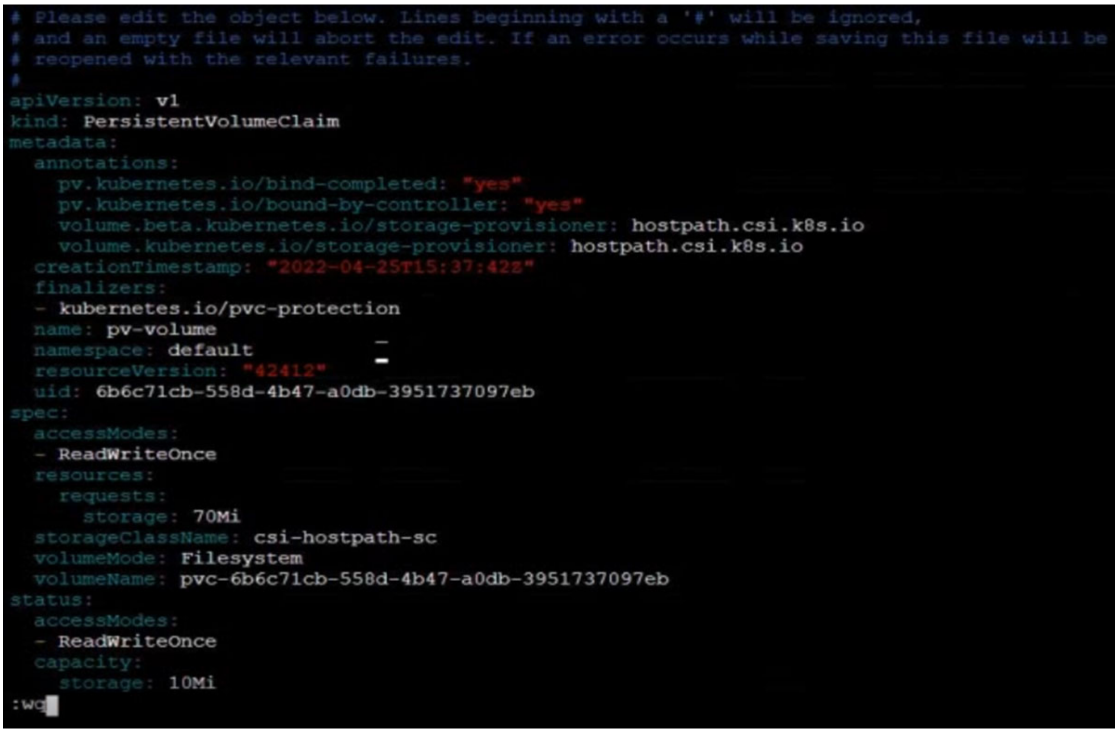
```
student@node-1:~$ kubectl config use-context ok8s
Switched to context "ok8s".
student@node-1:~$ vim pvc.yml
student@node-1:~$ kubectl get pv,pvc
No resources found
student@node-1:~$ kubectl create -f pvc.yml
persistentvolumeclaim/pv-volume created
student@node-1:~$ kubectl get pv,pvc
NAME                                                         CAPACITY   ACCESS MODES   RECLAIM POLICY    STATUS   CLAIM
         STORAGECLASS      REASON    AGE
persistentvolume/pvc-6b6c71cb-558d-4b47-a0db-3951737097eb   10Mi       RWO            Delete            Bound    default/pv-
volume   csi-hostpath-sc             3s

NAME                                STATUS    VOLUME                                     CAPACITY    ACCESS MODES   STORAGECLAS
S      AGE
persistentvolumeclaim/pv-volume    Bound    pvc-6b6c71cb-558d-4b47-a0db-3951737097eb   10Mi        RWO            csi-hostpat
h-sc   3s
student@node-1:~$
```



```
apiVersion: v1
kind: Pod
metadata:
  name: web-server
spec:
  volumes:
    - name: task-pv-storage
      persistentVolumeClaim:
        claimName: pv-volume
  containers:
    - name: web-server
      image: nginx
      volumeMounts:
        - mountPath: "/usr/share/nginx/html"
          name: task-pv-storage
```



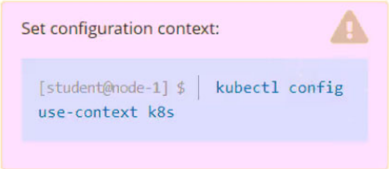
```
student@node-1:~$ kubectl config use-context ok8s
Switched to context "ok8s".
student@node-1:~$ vim pvc.yml
student@node-1:~$ kubectl get pv,pvc
No resources found
student@node-1:~$ kubectl create -f pvc.yml
persistentvolumeclaim/pv-volume created
student@node-1:~$ kubectl get pv,pvc
NAME                                                        CAPACITY   ACCESS MODES   RECLAIM POLICY   STATUS   CLAIM
         STORAGECLASS      REASON   AGE
persistentvolume/pvc-6b6c71cb-558d-4b47-a0db-3951737097eb   10Mi       RWO            Delete           Bound    default/pv-
volume   csi-hostpath-sc            3s

NAME                                STATUS   VOLUME                                     CAPACITY   ACCESS MODES   STORAGECLAS
S      AGE
persistentvolumeclaim/pv-volume     Bound    pvc-6b6c71cb-558d-4b47-a0db-3951737097eb   10Mi       RWO            csi-hostpat
h-sc   3s
student@node-1:~$
student@node-1:~$
student@node-1:~$ vim pod.yml
student@node-1:~$ kubectl create -f pod.yml
pod/web-server created
student@node-1:~$ kubectl get pods
NAME                    READY   STATUS              RESTARTS   AGE
csi-hostpath-socat-0    1/1     Running             0          6h12m
csi-hostpathplugin-0    9/9     Running             0          6h12m
web-server              0/1     ContainerCreating   0          5s
student@node-1:~$ kubectl get pods -w
NAME                    READY   STATUS              RESTARTS   AGE
csi-hostpath-socat-0    1/1     Running             0          6h12m
csi-hostpathplugin-0    9/9     Running             0          6h12m
web-server              0/1     ContainerCreating   0          10s
web-server              1/1     Running             0          22s
^Cstudent@node-1:~$ kubectl get pods
NAME                    READY   STATUS    RESTARTS   AGE
csi-hostpath-socat-0    1/1     Running   0          6h12m
csi-hostpathplugin-0    9/9     Running   0          6h12m
web-server              1/1     Running   0          27s
student@node-1:~$ kubectl edit pvc pv-volume
```



```
# Please edit the object below. Lines beginning with a '#' will be ignored,
# and an empty file will abort the edit. If an error occurs while saving this file will be
# reopened with the relevant failures.
#
apiVersion: v1
kind: PersistentVolumeClaim
metadata:
  annotations:
    pv.kubernetes.io/bind-completed: "yes"
    pv.kubernetes.io/bound-by-controller: "yes"
    volume.beta.kubernetes.io/storage-provisioner: hostpath.csi.k8s.io
    volume.kubernetes.io/storage-provisioner: hostpath.csi.k8s.io
  creationTimestamp: "2022-04-25T15:37:42Z"
  finalizers:
  - kubernetes.io/pvc-protection
  name: pv-volume
  namespace: default
  resourceVersion: "42412"
  uid: 6b6c71cb-558d-4b47-a0db-3951737097eb
spec:
  accessModes:
  - ReadWriteOnce
  resources:
    requests:
      storage: 70Mi
  storageClassName: csi-hostpath-sc
  volumeMode: Filesystem
  volumeName: pvc-6b6c71cb-558d-4b47-a0db-3951737097eb
status:
  accessModes:
  - ReadWriteOnce
  capacity:
    storage: 10Mi
:wq
```

Question 17 (Topic 1)

SIMULATION -



Set configuration context:

```
[student@node-1] $ | kubectl config
use-context k8s
```

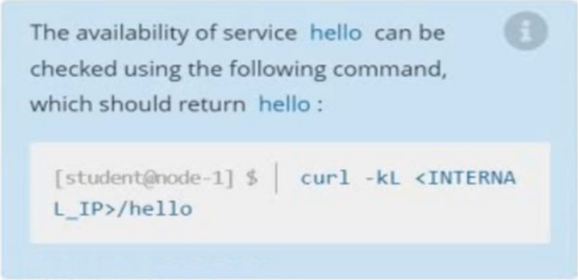
Task -

Create a new nginx Ingress resource as follows:

⇨ Name: pong

⇨ Namespace: ing-internal

⇨ Exposing service hello on path /hello using service port 5678

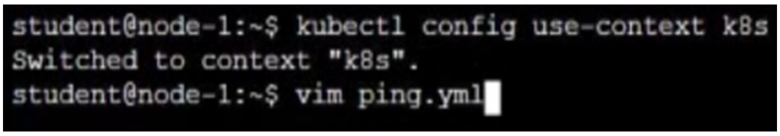


The availability of service hello can be checked using the following command, which should return hello :

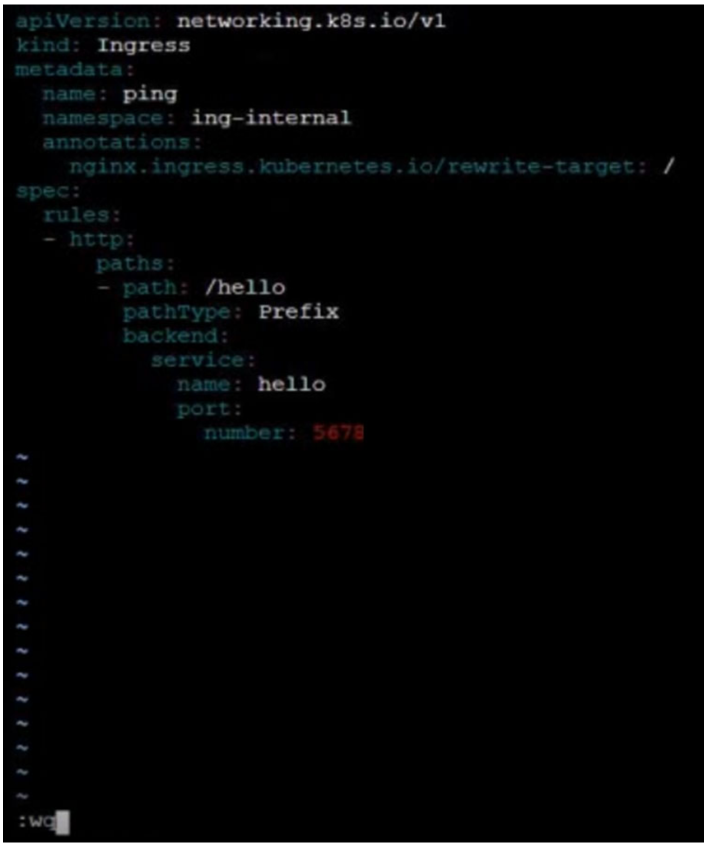
```
[student@node-1] $ | curl -kL <INTERNA
L_IP>/hello
```

Answer : **See explanation below.**

Explanation:



```
student@node-1:~$ kubectl config use-context k8s
Switched to context "k8s".
student@node-1:~$ vim ping.yml
```



```
apiVersion: networking.k8s.io/v1
kind: Ingress
metadata:
  name: ping
  namespace: ing-internal
  annotations:
    nginx.ingress.kubernetes.io/rewrite-target: /
spec:
  rules:
  - http:
      paths:
      - path: /hello
        pathType: Prefix
        backend:
          service:
            name: hello
            port:
              number: 5678
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
:wq
```

Page: 4 / 4
Total 20 questions

Previous Page

10 questions per page