

CKA^{Q&As}

Certified Kubernetes Administrator (CKA) Program

Pass Linux Foundation CKA Exam with 100% Guarantee

Free Download Real Questions & Answers **PDF** and **VCE** file from:

<https://www.leads4pass.com/cka.html>

100% Passing Guarantee
100% Money Back Assurance

Following Questions and Answers are all new published by Linux Foundation Official Exam Center

-  **Instant Download** After Purchase
-  **100% Money Back** Guarantee
-  **365 Days** Free Update
-  **800,000+** Satisfied Customers



QUESTION 1

CORRECT TEXT

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

Correct Answer: Check the answer in explanation.

```
kubect1 get pods -o wide
```

```
kubectl delete po "nginx-dev" kubectl delete po "nginx-prod"
```

QUESTION 2

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

Correct Answer: Check the answer in explanation.

Solution

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

QUESTION 3

Get list of all pods in all namespaces and write it to file "/opt/pods-list.yaml"

Correct Answer: Check the answer in explanation.

```
kubectl get po -all-namespaces > /opt/pods-list.yaml
```

QUESTION 4

Create a pod that echo "hello world" and then exists. Have the pod deleted automatically when it's completed

Correct Answer: Check the answer in explanation.

```
kubectl run busybox --image=busybox -it --rm --restart=Never -- /bin/sh -c '\echo hello world\' kubectl get po # You shouldn't see pod with the name "busybox"
```

QUESTION 5

CORRECT TEXT



Task

Scale the deployment presentation to 6 pods.

Correct Answer: Check the answer in explanation.

```
kubectl get deployment kubectl scale deployment.apps/presentation --replicas=6
```

QUESTION 6

List the nginx pod with custom columns POD_NAME and POD_STATUS

Correct Answer: Check the answer in explanation.

Solution

```
kubectl get po -o=custom-columns="POD_NAME:.metadata.name, POD_STATUS:.status.containerStatuses[].state"
```

QUESTION 7

SIMULATION For this item, you will have to ssh to the nodes ik8s-master-0 and ik8s-node-0 and complete all tasks on these nodes. Ensure that you return to the base node (hostname: node-1) when you have completed this item. Context

As an administrator of a small development team, you have been asked to set up a Kubernetes cluster to test the viability of a new application.

Task

You must use kubeadm to perform this task. Any kubeadm invocations will require the use of the -- ignore-preflight-errors=all option.

Configure the node ik8s-master-0 as a master node. .

Join the node ik8s-node-0 to the cluster.

Correct Answer: Check the answer in explanation.

Solution

You must use the kubeadm configuration file located at `/etc/kubeadm.conf` when initializing your cluster.

You may use any CNI plugin to complete this task, but if you don't have your favourite CNI plugin's manifest URL at hand, Calico is one popular option:

<https://docs.projectcalico.org/v3.14/manifests/calico.yaml> Docker is already installed on both nodes and apt has been configured so that you can install the required tools.

QUESTION 8

SIMULATION

Given a partially-functioning Kubernetes cluster, identify symptoms of failure on the cluster.

Determine the node, the failing service, and take actions to bring up the failed service and restore the health of the cluster. Ensure that any changes are made permanently.

You can ssh to the relevant nodes (bk8s-master-0 or bk8s-node-0) using:

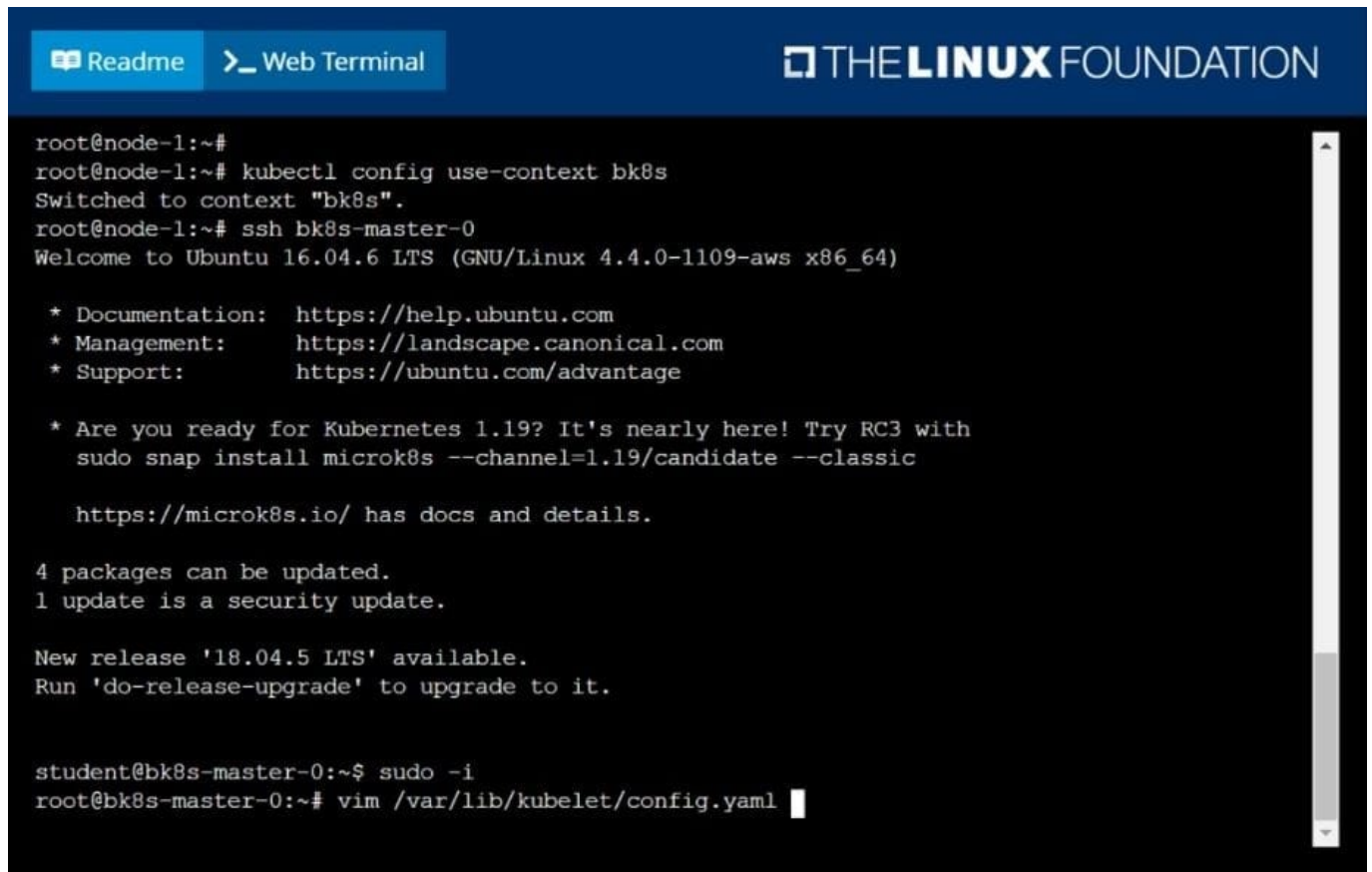
```
[student@node-1] $ ssh
```

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

```
[student@nodename] $ | sudo -i
```

Correct Answer: Check the answer in explanation.

solution



The screenshot shows a web terminal interface with a dark background and white text. At the top, there are two tabs: 'Readme' and 'Web Terminal'. The 'Web Terminal' tab is active. In the top right corner, the 'THE LINUX FOUNDATION' logo is visible. The terminal content shows a user logging into a node, switching the kubectl context to 'bk8s', and then SSHing into a master node. The master node is an Ubuntu 16.04.6 LTS system. The terminal output includes system messages about updates and a new release, followed by the user running 'sudo -i' to become root and then 'vim /var/lib/kubelet/config.yaml'.

```
root@node-1:~#
root@node-1:~# kubectl config use-context bk8s
Switched to context "bk8s".
root@node-1:~# ssh bk8s-master-0
Welcome to Ubuntu 16.04.6 LTS (GNU/Linux 4.4.0-1109-aws x86_64)

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

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

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

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

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

student@bk8s-master-0:~$ sudo -i
root@bk8s-master-0:~# vim /var/lib/kubelet/config.yaml
```

```
Readme Web Terminal THE LINUX FOUNDATION

authorization:
  mode: Webhook
  webhook:
    cacheAuthorizedTTL: 0s
    cacheUnauthorizedTTL: 0s
clusterDNS:
- 10.96.0.10
clusterDomain: cluster.local
cpuManagerReconcilePeriod: 0s
evictionPressureTransitionPeriod: 0s
fileCheckFrequency: 0s
healthzBindAddress: 127.0.0.1
healthzPort: 10248
httpCheckFrequency: 0s
imageMinimumGCAge: 0s
kind: KubeletConfiguration
nodeStatusReportFrequency: 0s
nodeStatusUpdateFrequency: 0s
rotateCertificates: true
runtimeRequestTimeout: 0s
staticPodPath: /etc/kubernetes/manifests
streamingConnectionIdleTimeout: 0s
syncFrequency: 0s
volumeStatsAggPeriod: 0s
:Wg
```

```
Readme Web Terminal THE LINUX FOUNDATION

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

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

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

student@bk8s-master-0:~$ sudo -i
root@bk8s-master-0:~# vim /var/lib/kubelet/config.yaml
root@bk8s-master-0:~# systemctl restart kubelet
root@bk8s-master-0:~# systemctl enable kubelet
root@bk8s-master-0:~# kubectl get nodes

NAME             STATUS    ROLES    AGE   VERSION
bk8s-master-0   Ready    master   77d   v1.18.2
bk8s-node-0     Ready    <none>   77d   v1.18.2
root@bk8s-master-0:~#
root@bk8s-master-0:~# exit
logout
student@bk8s-master-0:~$ exit
logout
Connection to 10.250.4.77 closed.
root@node-1:~#
```

QUESTION 9

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

Correct Answer: Check the answer in 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 -n engineering -f YAML File:
```

```
apiVersion: v1 kind: Pod metadata: name: nginx namespace: engineering labels: env: test spec: containers:
```

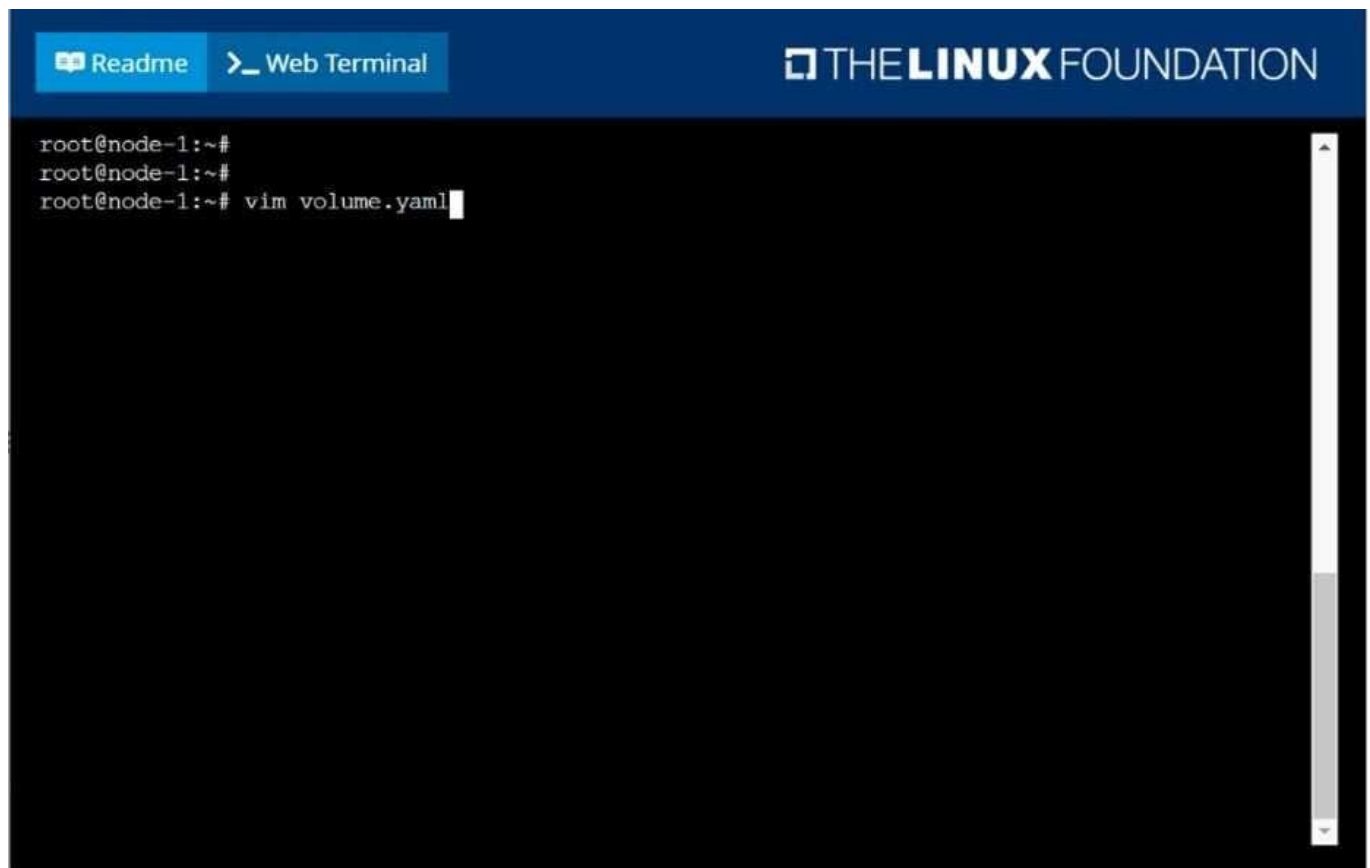
```
-name: nginx image: nginx imagePullPolicy: IfNotPresent restartPolicy: Never
```

```
kubectl create -f nginx-pod.yaml
```

QUESTION 10**SIMULATION**

Create a pod as follows: Name: non-persistent-redis container Image: redis Volume with name: cache-control Mount path: /data/redis The pod should launch in the staging namespace and the volume must not be persistent.

Correct Answer: Check the answer in explanation.



The screenshot shows a web terminal interface with a dark background. At the top, there are two tabs: 'Readme' and 'Web Terminal'. The 'Web Terminal' tab is active. In the top right corner, the logo for 'THE LINUX FOUNDATION' is visible. The terminal content shows a root user at a node-1 prompt, followed by three lines of shell commands: 'root@node-1:~#', 'root@node-1:~#', and 'root@node-1:~# vim volume.yaml'. A vertical scrollbar is visible on the right side of the terminal window.

```
Readme Web Terminal THE LINUX FOUNDATION

apiVersion: v1
kind: Pod
metadata:
  name: non-persistent-redis
  namespace: staging
spec:
  containers:
  - name: redis
    image: redis
    volumeMounts:
    - name: cache-control
      mountPath: /data/redis
  volumes:
  - name: cache-control
    emptyDir: {}
~
~
~
~
~
~
~
~
~
:w
```

```
Readme Web Terminal THE LINUX FOUNDATION

root@node-1:~#
root@node-1:~#
root@node-1:~# vim volume.yaml
root@node-1:~# k create -f volume.yaml
pod/non-persistent-redis created
root@node-1:~# k get po -n staging
NAME                READY   STATUS    RESTARTS   AGE
non-persistent-redis 1/1     Running   0           6s
root@node-1:~#
```


QUESTION 11

List the nginx pod with custom columns POD_NAME and POD_STATUS

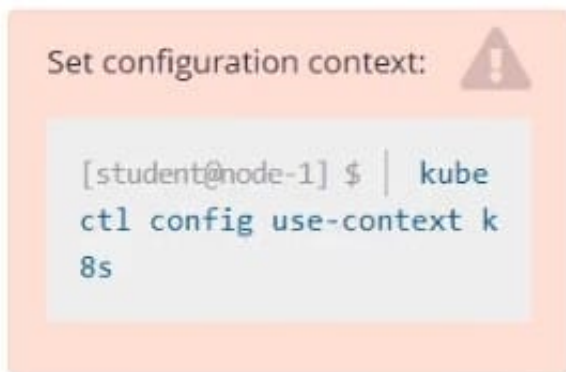
Correct Answer: Check the answer in explanation.

Solution

```
kubectl get po -o=custom-columns="POD_NAME:.metadata.name, POD_STATUS:.status.containerStatuses[].state"
```

QUESTION 12

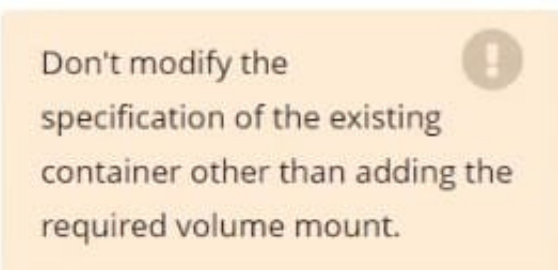
CORRECT TEXT



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-corp-app.log
```

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

Correct Answer: Check the answer in explanation.

```
# kubectl get pod big-corp-app -o yaml # apiVersion: v1 kind: Pod metadata: name: big-corp-app spec: containers:
```

```
-name: big-corp-app
```

image: busybox

args:

-/bin/sh

- -c

->

i=0;

while true;

do

echo "\$(date) INFO \$i" >> /var/log/big-corp-app.log; i=\$((i+1));

sleep 1;

done

volumeMounts:

-

name: logs

mountPath: /var/log

-

name: count-log-1

image: busybox

args: [/bin/sh, -c, \\'tail -n+1 -f /var/log/big-corp-app.log\\'] volumeMounts:

-

name: logs

mountPath: /var/log

volumes:

-

name: logs

emptyDir: {

}

kubectl logs big-corp-app -c count-log-1

QUESTION 13

CORRECT TEXT



Task

Schedule a pod as follows:

1.

Name: nginx-kusc00401

2.

Image: nginx

3.

Node selector: disk=ssd

Correct Answer: Check the answer in explanation.

Solution:

#yaml apiVersion: v1 kind: Pod metadata: name: nginx-kusc00401 spec: containers:

`-name: nginx image: nginx imagePullPolicy: IfNotPresent nodeSelector: disk: spinning # kubectl create -f node-select.yaml`**QUESTION 14**Create a namespace called `development` and a pod with image `nginx` called `nginx` on this namespace.

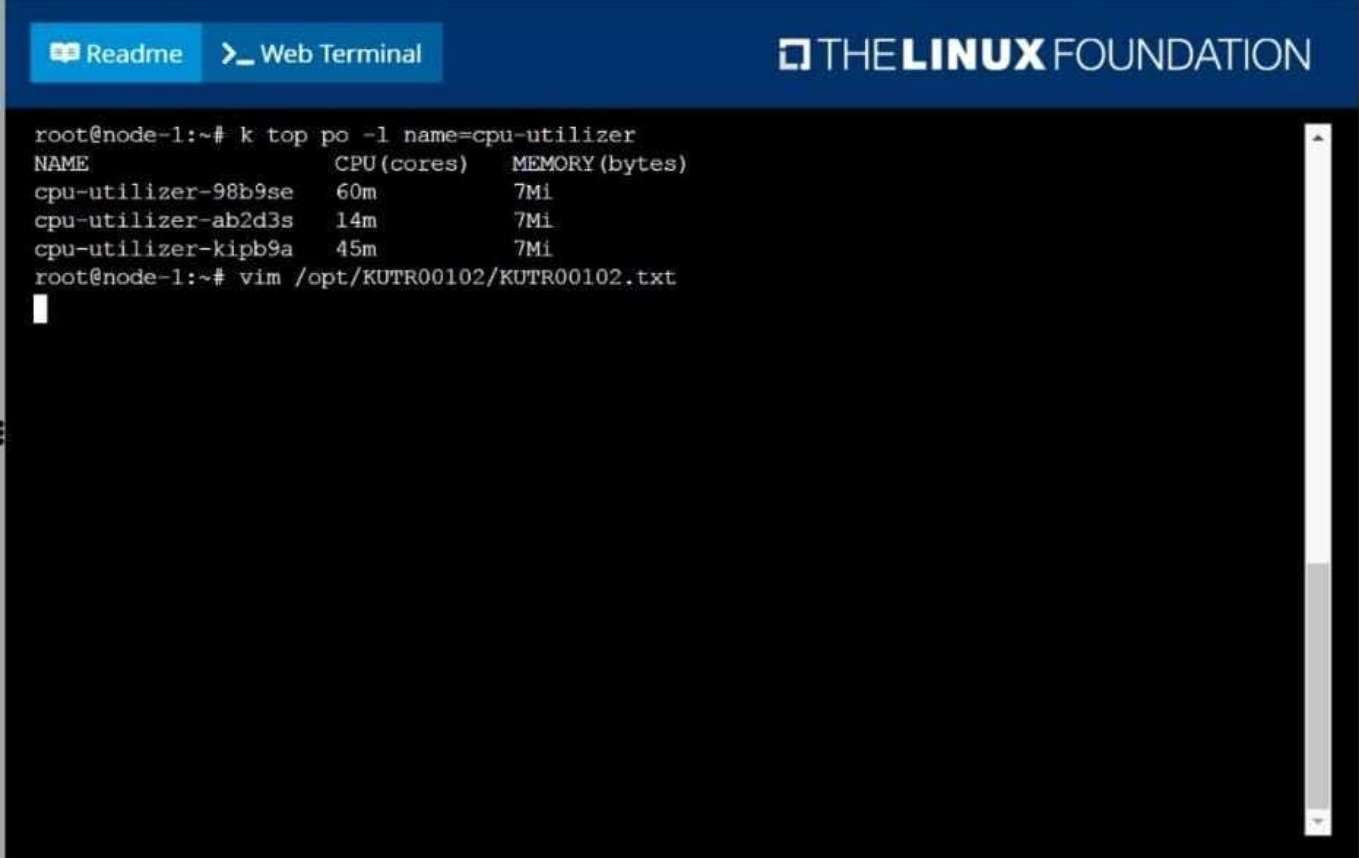
Correct Answer: Check the answer in explanation.

`kubectl create namespace development kubectl run nginx --image=nginx --restart=Never -n development`**QUESTION 15**

SIMULATION

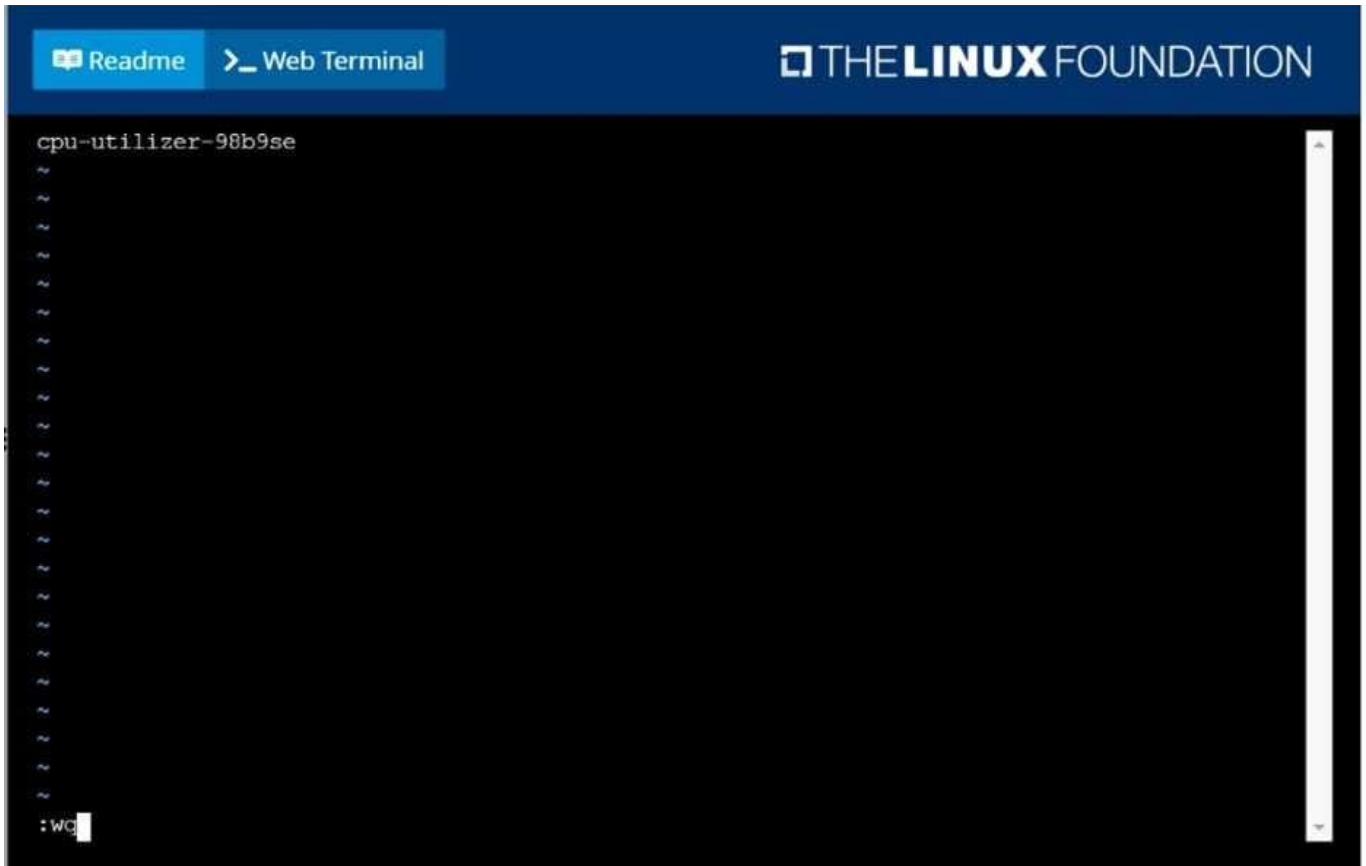
From the pod label name=cpu-utilizer, find pods running high CPU workloads and write the name of the pod consuming most CPU to the file /opt/KUTR00102/KUTR00102.txt (which already exists).

Correct Answer: Check the answer in explanation.



The screenshot shows a web terminal interface with a dark background. At the top, there are two tabs: 'Readme' and 'Web Terminal'. The 'Web Terminal' tab is active. In the top right corner, the logo for 'THE LINUX FOUNDATION' is visible. The terminal content shows a user running a command to list pods with high CPU usage. The output shows three pods, with 'cpu-utilizer-98b9se' having the highest CPU usage at 60m.

```
root@node-1:~# k top po -l name=cpu-utilizer
NAME                CPU(cores)  MEMORY(bytes)
cpu-utilizer-98b9se  60m         7Mi
cpu-utilizer-ab2d3s  14m         7Mi
cpu-utilizer-kipb9a  45m         7Mi
root@node-1:~# vim /opt/KUTR00102/KUTR00102.txt
█
```



[Latest CKA Dumps](#)

[CKA PDF Dumps](#)

[CKA Brindumps](#)