Linux-Foundation

Exam Questions CKAD

Certified Kubernetes Application Developer (CKAD) Program

About Exambible

Your Partner of IT Exam

Found in 1998

Exambible is a company specialized on providing high quality IT exam practice study materials, especially Cisco CCNA, CCDA, CCNP, CCIE, Checkpoint CCSE, CompTIA A+, Network+ certification practice exams and so on. We guarantee that the candidates will not only pass any IT exam at the first attempt but also get profound understanding about the certificates they have got. There are so many alike companies in this industry, however, Exambible has its unique advantages that other companies could not achieve.

Our Advances

* 99.9% Uptime

    All examinations will be up to date.

* 24/7 Quality Support

    We will provide service round the clock.
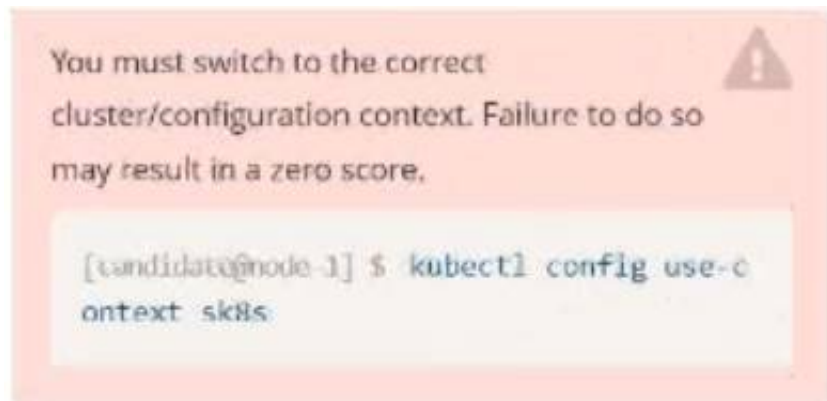
* 100% Pass Rate

    Our guarantee that you will pass the exam.

* Unique Gurantee

    If you do not pass the exam at the first time, we will not only arrange FULL REFUND for you, but also provide you another exam of your claim, ABSOLUTELY FREE!

NEW QUESTION 1
Exhibit:

> You must switch to the correct cluster/configuration context. Failure to do so may result in a zero score.
>
> [candidate@node-1] $ kubectl config use-context sk8s

Task:
Key3: value1
Add an environment variable named BEST_VARIABLE consuming the value of the secret key3.

A. Mastered
B. Not Mastered

Answer: A

Explanation:
 Solution:

```
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
```

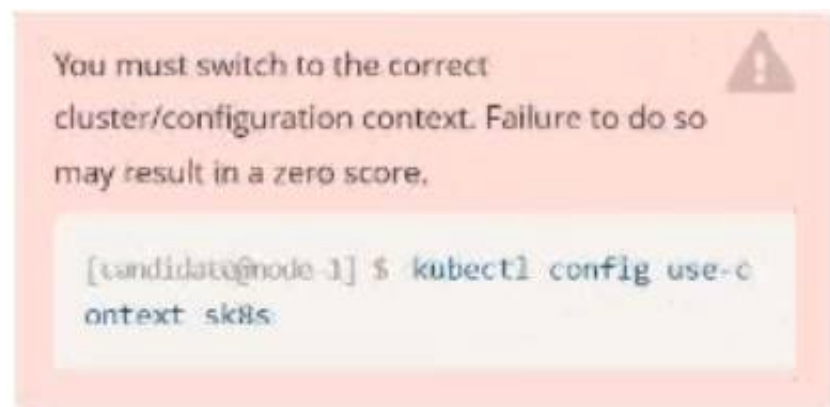Text Description automatically generated

```
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
```

Text Description automatically generated

```
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:~$
```

NEW QUESTION 2
Exhibit:

You must switch to the correct cluster/configuration context. Failure to do so may result in a zero score.

```
[candidate@node-1] $ kubectl config use-context sk8s
```

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

A. Mastered
B. Not Mastered

Answer: A

Explanation:

Solution:

```
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:~$
```

Text Description automatically generated

```
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
```

Text Description automatically generated

```
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:~$
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:~$
```

NEW QUESTION 3
Exhibit:

Set configuration context:

```
[student@node-1] $ | kubectl config
use-context nk8s
```

Task
A deployment is falling on the cluster due to an incorrect image being specified. Locate the deployment, and fix the problem.
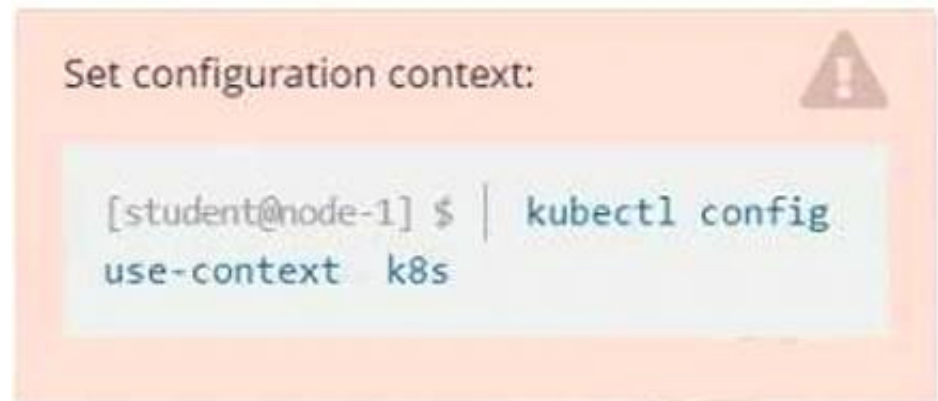
A. Mastered
B. Not Mastered

Answer: A

Explanation:
create deploy hello-deploy --image=nginx --dry-run=client -o yaml > hello-deploy.yaml
Update deployment image to nginx:1.17.4: kubec
nginx=nginx:1.17.4

NEW QUESTION 4
Exhibit:

Set configuration context:

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

Context
You sometimes need to observe a pod's logs, and write those logs to a file for further analysis. Task
Please complete the following;
• Deploy the counter pod to the cluster using the provided YAMLspec file at /opt/KDOB00201/counter.yaml
• Retrieve all currently available application logs from the running pod and store them in the file
/opt/KDOB0020l/log_Output.txt, which has already been created

A. Mastered
B. Not Mastered

Answer: A

Explanation:

Solution:

```
student@node-1:~$ kubectl create -f /opt/KDOB00201/counter.yaml
pod/counter created
student@node-1:~$ kubectl get pods
NAME                READY   STATUS    RESTARTS   AGE
counter             1/1     Running   0          10s
liveness-http       1/1     Running   0          6h45m
nginx-101           1/1     Running   0          6h46m
nginx-configmap     1/1     Running   0          107s
nginx-secret        1/1     Running   0          7m21s
poller              1/1     Running   0          6h46m
student@node-1:~$ kubectl logs counter
1: 2b305101817ae25ca60ae46510fb6d11
2: 3648cf2eae95ab680dba8f195f891af4
3: 65c8bbd4dbf70bf81f2a0984a3a44ede
4: 40d3a9c8e46f5533bb4828fbe5c8d038
5: 390442d2530a90c3602901e3fe999ac8
6: b71d95187417e139effb33af77681040
7: 66a8e55a6491e756d2d0549ad6ab90a7
8: ff2b3d583b64125d2f9129c443bb37ff
9: b6c6a12b6e77944ed8baaaf6c242dae4
10: bfcc9a894a0604fc4b814b37d0a200a4
student@node-1:~$ kubectl logs counter  > /opt/KDOB00201/log_output.txt
student@node-1:~$ []
```
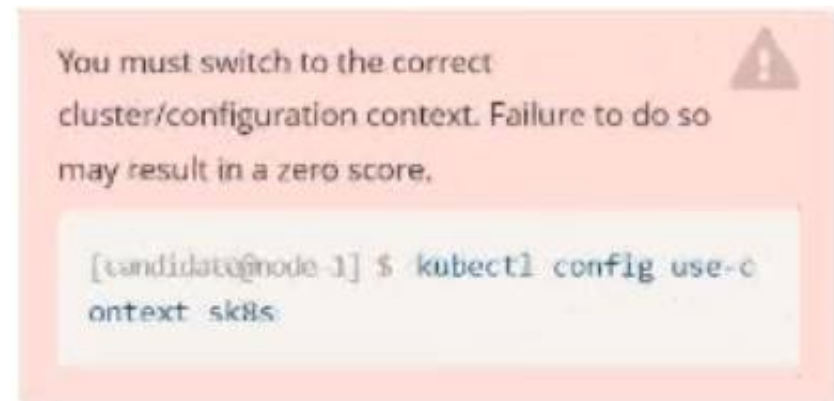
```
student@node-1:~$ kubectl logs counter  > /opt/KDOB00201/log_output.txt
student@node-1:~$ kubectl logs counter  > /opt/KDOB00201/log_output.txt
student@node-1:~$ cat /opt/KDOB00201/log_output.txt
```

```
📖 Readme   >_ Web Terminal            □ THE LINUX FOUNDATION

student@node-1:~$ kubectl logs counter  > /opt/KDOB00201/log_output.txt
student@node-1:~$ cat /opt/KDOB00201/log_output.txt
1: 2b305101817ae25ca60ae46510fb6d11
2: 3648cf2eae95ab680dba8f195f891af4
3: 65c8bbd4dbf70bf81f2a0984a3a44ede
4: 40d3a9c8e46f5533bb4828fbe5c8d038
5: 390442d2530a90c3602901e3fe999ac8
6: b71d95187417e139effb33af77681040
7: 66a8e55a6491e756d2d0549ad6ab90a7
8: ff2b3d583b64125d2f9129c443bb37ff
9: b6c6a12b6e77944ed8baaaf6c242dae4
10: bfcc9a894a0604fc4b814b37d0a200a4
11: 5493cd16a1790a5fb9512b0c9d4c5dd1
12: 03f169e93e6143438e6dfe4ecb3cc9ed
13: 764b37fe611373c42d0b47154041f6eb
14: 1a56fbe1896b0ee6394136166281839e
15: ecc492eb17715de090c47345a98d98d3
16: 7974a6bec0fb44b6b8bbfc71aa3fbe74
17: 9ae01bef01748b12cc9f97a5f9f72cd6
18: 23fb22ee34d4272e4c9e005f1774515f
19: ec7e1a5d314da9a0ad45d53be5a7acae
20: 0bccdd8ee02cd42029e8162cd1c1197c
21: d6851ea43546216b95bcb81ced997102
22: 7ed9a38ea8bf0d86206569481442af44
23: 29b8416ddc63dbfcb987ab3c8198e9fe
24: 1f2062001df51a108ab25010f506716f
student@node-1:~$
```

NEW QUESTION 5
Exhibit:

> You must switch to the correct
> cluster/configuration context. Failure to do so
> may result in a zero score.
>
> [candidate@node-1] $ kubectl config use-c
> ontext sk8s

Task:
≫ To run 2 replicas of the pod
≫ Add the following label on the pod:
Role userUI

A. Mastered
B. Not Mastered

Answer: A

Explanation:

Solution:
Text Description automatically generated

```
File Edit View Terminal Tabs Help
# reopened with the relevant failures.
#
apiVersion: apps/v1
kind: Deployment
metadata:
  annotations:
    deployment.kubernetes.io/revision: "1"
  creationTimestamp: "2022-09-24T04:27:03Z"
  generation: 1
  labels:
    app: nginx
  name: ckad00017-deployment
  namespace: ckad00017
  resourceVersion: "3349"
  uid: 1cd67613-fade-46e9-b741-94298b9c6e7c
spec:
  progressDeadlineSeconds: 600
  replicas: 2
  revisionHistoryLimit: 10
  selector:
    matchLabels:
      app: nginx
  strategy:
    rollingUpdate:
      maxSurge: 25%
      maxUnavailable: 25%
    type: RollingUpdate
  template:
    metadata:
      creationTimestamp: null
      labels:
-- INSERT --                                    33,14          5%
```

Text Description automatically generated

```
File Edit View Terminal Tabs Help
  name: ckad00017-deployment
  namespace: ckad00017
  resourceVersion: "3349"
  uid: 1cd67613-fade-46e9-b741-94298b9c6e7c
spec:
  progressDeadlineSeconds: 600
  replicas: 2
  revisionHistoryLimit: 10
  selector:
    matchLabels:
      app: nginx
  strategy:
    rollingUpdate:
      maxSurge: 25%
      maxUnavailable: 25%
    type: RollingUpdate
  template:
    metadata:
      creationTimestamp: null
      labels:
        app: nginx
        role: userUI
    spec:
      containers:
      - image: nginx:latest
        imagePullPolicy: Always
        name: nginx
        ports:
        - containerPort: 80
          protocol: TCP
        resources: {}
-- INSERT --                                    35,21          33%
```

Text Description automatically generated

```
File Edit View Terminal Tabs Help
backend-deployment-59d449b99d-h2zjq    0/1    Running    0    9s
backend-deployment-78976f74f5-b8c85    1/1    Running    0    6h40m
backend-deployment-78976f74f5-flfsj    1/1    Running    0    6h40m
candidate@node-1:~$ kubectl get deploy -n staging
NAME                READY    UP-TO-DATE    AVAILABLE    AGE
backend-deployment  3/3      3             3            6h40m
candidate@node-1:~$ kubectl get deploy -n staging
NAME                READY    UP-TO-DATE    AVAILABLE    AGE
backend-deployment  3/3      3             3            6h41m
candidate@node-1:~$ vim ~/spicy-pikachu/backend-deployment.yaml
candidate@node-1:~$ kubectl config use-context k8s
Switched to context "k8s".
candidate@node-1:~$ kubectl  set serviceaccount  deploy app-1 app -n frontend
deployment.apps/app-1 serviceaccount updated
candidate@node-1:~$ kubectl config use-context k8s
Switched to context "k8s".
candidate@node-1:~$ vim ~/prompt-escargot/buffalo-deployment.yaml
candidate@node-1:~$ vim ~/prompt-escargot/buffalo-deployment.yaml
candidate@node-1:~$ kubectl  apply  -f ~/prompt-escargot/buffalo-deployment.yaml
deployment.apps/buffalo-deployment configured
candidate@node-1:~$ kubectl  get pods -n gorilla
NAME                                   READY    STATUS             RESTARTS    AGE
buffalo-deployment-776844df7f-r5fsb    1/1      Running            0           6h38m
buffalo-deployment-859898c6f5-zx5gj    0/1      ContainerCreating  0           8s
candidate@node-1:~$ kubectl  get deploy -n gorilla
NAME                READY    UP-TO-DATE    AVAILABLE    AGE
buffalo-deployment  1/1      1             1            6h38m
candidate@node-1:~$ kubectl config use-context k8s
Switched to context "k8s".
candidate@node-1:~$ kubectl edit deploy ckad00017-deployment -n ckad00017
deployment.apps/ckad00017-deployment edited
candidate@node-1:~$
```

```
File Edit View Terminal Tabs Help
candidate@node-1:~$ kubectl  get pods -n gorilla
NAME                                   READY    STATUS             RESTARTS    AGE
buffalo-deployment-776844df7f-r5fsb    1/1      Running            0           6h38m
buffalo-deployment-859898c6f5-zx5gj    0/1      ContainerCreating  0           8s
candidate@node-1:~$ kubectl  get deploy -n gorilla
NAME                READY    UP-TO-DATE    AVAILABLE    AGE
buffalo-deployment  1/1      1             1            6h38m
candidate@node-1:~$ kubectl config use-context k8s
Switched to context "k8s".
candidate@node-1:~$ kubectl edit deploy ckad00017-deployment -n ckad00017
deployment.apps/ckad00017-deployment edited
candidate@node-1:~$ kubectl  expose  deploy ckad00017-deployment -n ckad0001
ckad00014  ckad00015  ckad00017
candidate@node-1:~$ kubectl  expose  deploy ckad00017-deployment -n ckad0001
ckad00014  ckad00015  ckad00017
candidate@node-1:~$ kubectl  expose  deploy ckad00017-deployment -n ckad0001
ckad00014  ckad00015  ckad00017
candidate@node-1:~$ kubectl  expose  deploy ckad00017-deployment -n ckad0001
ckad00014  ckad00015  ckad00017
candidate@node-1:~$ kubectl  expose  deploy ckad00017-deployment -n ckad0001
ckad00014  ckad00015  ckad00017
candidate@node-1:~$ kubectl  expose  deploy ckad00017-deployment -n ckad0001
ckad00014  ckad00015  ckad00017
candidate@node-1:~$ kubectl  expose  deploy ckad00017-deployment -n ckad0001
ckad00014  ckad00015  ckad00017
candidate@node-1:~$ kubectl  expose  deploy ckad00017-deployment -n ckad0001
ckad00014  ckad00015  ckad00017
candidate@node-1:~$ kubectl  expose  deploy ckad00017-deployment -n ckad0001
ckad00014  ckad00015  ckad00017
candidate@node-1:~$ kubectl  expose  deploy ckad00017-deployment -n ckad00017 --name=cherry --port=8888 --type=NodePort
service/cherry exposed
candidate@node-1:~$
```

```
candidate@node-1:~$ kubectl get svc
NAME         TYPE         CLUSTER-IP     EXTERNAL-IP    PORT(S)    AGE
kubernetes   ClusterIP    10.96.0.1      <none>         443/TCP    77d
candidate@node-1:~$ kubectl get svc  -n ckad00017
NAME     TYPE       CLUSTER-IP       EXTERNAL-IP    PORT(S)          AGE
cherry   NodePort   10.100.100.176   <none>         8888:30683/TCP   24s
candidate@node-1:~$ kubectl  expose  service  deploy ckad00017-deployment -n ckad00017 --name=cherry --port=8888 --type=N
odePort
Error from server (NotFound): services "deploy" not found
Error from server (NotFound): services "ckad00017-deployment" not found
candidate@node-1:~$ kubectl get svc  -n ckad00017
NAME     TYPE       CLUSTER-IP       EXTERNAL-IP    PORT(S)          AGE
cherry   NodePort   10.100.100.176   <none>         8888:30683/TCP   46s
candidate@node-1:~$
```

```
File Edit View Terminal Tabs Help
candidate@node-1:~$ kubectl expose service  deploy ckad00017-deployment -n ckad00017 --name=cherry --port=8888 --type=N
odePort
Error from server (NotFound): services "deploy" not found
Error from server (NotFound): services "ckad00017-deployment" not found
candidate@node-1:~$ kubectl get svc  -n ckad00017
NAME     TYPE        CLUSTER-IP       EXTERNAL-IP    PORT(S)        AGE
cherry   NodePort    10.100.100.176   <none>         8888:30683/TCP   46s
candidate@node-1:~$ history
    1  vi ~/spicy-pikachu/backend-deployment.yaml
    2  kubectl config use-context sk8s
    3  vim .vimrc
    4  vim ~/spicy-pikachu/backend-deployment.yaml
    5  kubectl apply  -f ~/spicy-pikachu/backend-deployment.yaml
    6  kubectl get  pods -n staging
    7  kubectl get deploy -n staging
    8  vim ~/spicy-pikachu/backend-deployment.yaml
    9  kubectl config use-context k8s
   10  kubectl set serviceaccount  deploy app-1 app -n frontend
   11  kubectl config use-context k8s
   12  vim ~/prompt-escargot/buffalo-deployment.yaml
   13  kubectl apply  -f ~/prompt-escargot/buffalo-deployment.yaml
   14  kubectl get pods -n gorilla
   15  kubectl  get deploy -n gorilla
   16  kubectl config use-context k8s
   17  kubectl edit deploy ckad00017-deployment -n ckad00017
   18  kubectl expose  deploy ckad00017-deployment -n ckad00017 --name=cherry --port=8888 --type=NodePort
   19  kubectl get svc
   20  kubectl get svc  -n ckad00017
   21  kubectl expose  service  deploy ckad00017-deployment -n ckad00017 --name=cherry --port=8888 --type=NodePort
   22  kubectl get svc  -n ckad00017
   23  history
candidate@node-1:~$
```

NEW QUESTION 6
Exhibit:

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

Context
A container within the poller pod is hard-coded to connect the nginxsvc service on port 90. As this port changes to 5050 an additional container needs to be added to the poller pod which adapts the container to connect to this new port. This should be realized as an ambassador container within the pod.
Task
• Update the nginxsvc service to serve on port 5050.
• Add an HAproxy container named haproxy bound to port 90 to the poller pod and deploy the enhanced pod. Use the image haproxy and inject the configuration located at /opt/KDMC00101/haproxy.cfg, with a ConfigMap named haproxy-config, mounted into the container so that haproxy.cfg is available at /usr/local/etc/haproxy/haproxy.cfg. Ensure that you update the args of the poller container to connect to
localhost instead of nginxsvc so that the connection is correctly proxied to the new service endpoint. You must not modify the port of the endpoint in poller's args . The spec file used to create the initial poller pod is available in /opt/KDMC00101/poller.yaml

A. Mastered
B. Not Mastered

Answer: A

Explanation:

Solution:
apiVersion: apps/v1 kind: Deployment metadata:
name: my-nginx spec:
selector:
matchLabels: run: my-nginx replicas: 2 template: metadata: labels:
run: my-nginx spec: containers:
- name: my-nginx image: nginx ports:
- containerPort: 90
This makes it accessible from any node in your cluster. Check the nodes the Pod is running on: kubectl apply -f ./run-my-nginx.yaml
kubectl get pods -l run=my-nginx -o wide
NAME READY STATUS RESTARTS AGE IP NODE
my-nginx-3800858182-jr4a2 1/1 Running 0 13s 10.244.3.4 kubernetes-minion-905m my-nginx-3800858182-kna2y 1/1 Running 0 13s 10.244.2.5 kubernetes-minion-ljyd Check your pods' IPs:
kubectl get pods -l run=my-nginx -o yaml | grep podIP podIP: 10.244.3.4
podIP: 10.244.2.5

NEW QUESTION 7
Exhibit:

Set configuration context:

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

Context
Developers occasionally need to submit pods that run periodically. Task
Follow the steps below to create a pod that will start at a predetermined time and]which runs to completion only once each time it is started:
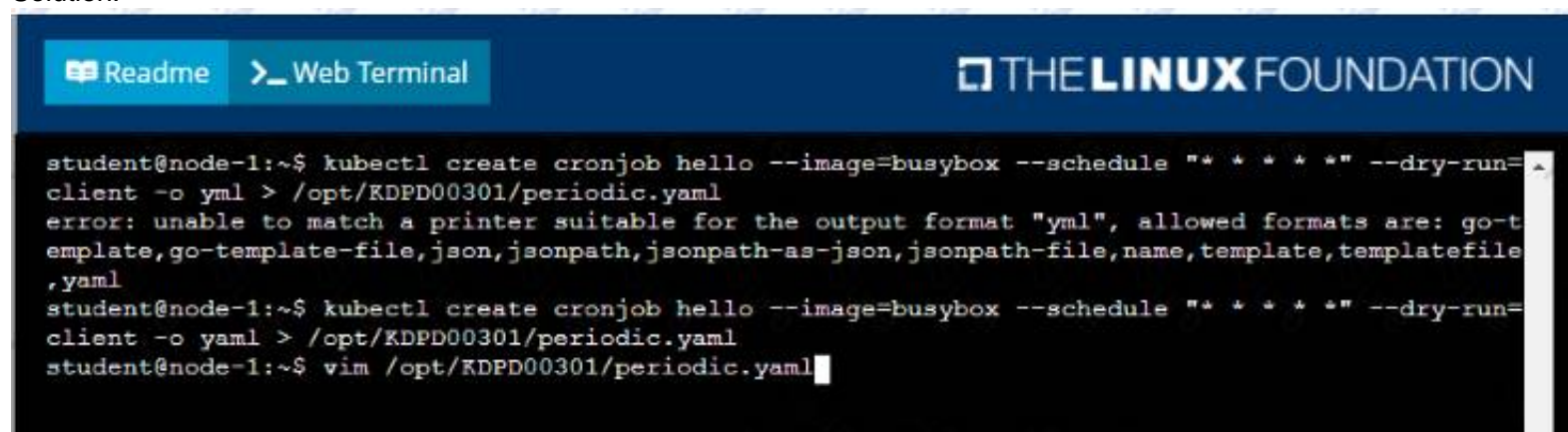• Create a YAML formatted Kubernetes manifest /opt/KDPD00301/periodic.yaml that runs the following shell command: date in a single busybox container. The command should run every minute and must complete within 22 seconds or be terminated oy Kubernetes. The Cronjob namp and container name should both be hello
• Create the resource in the above manifest and verify that the job executes successfully at least once

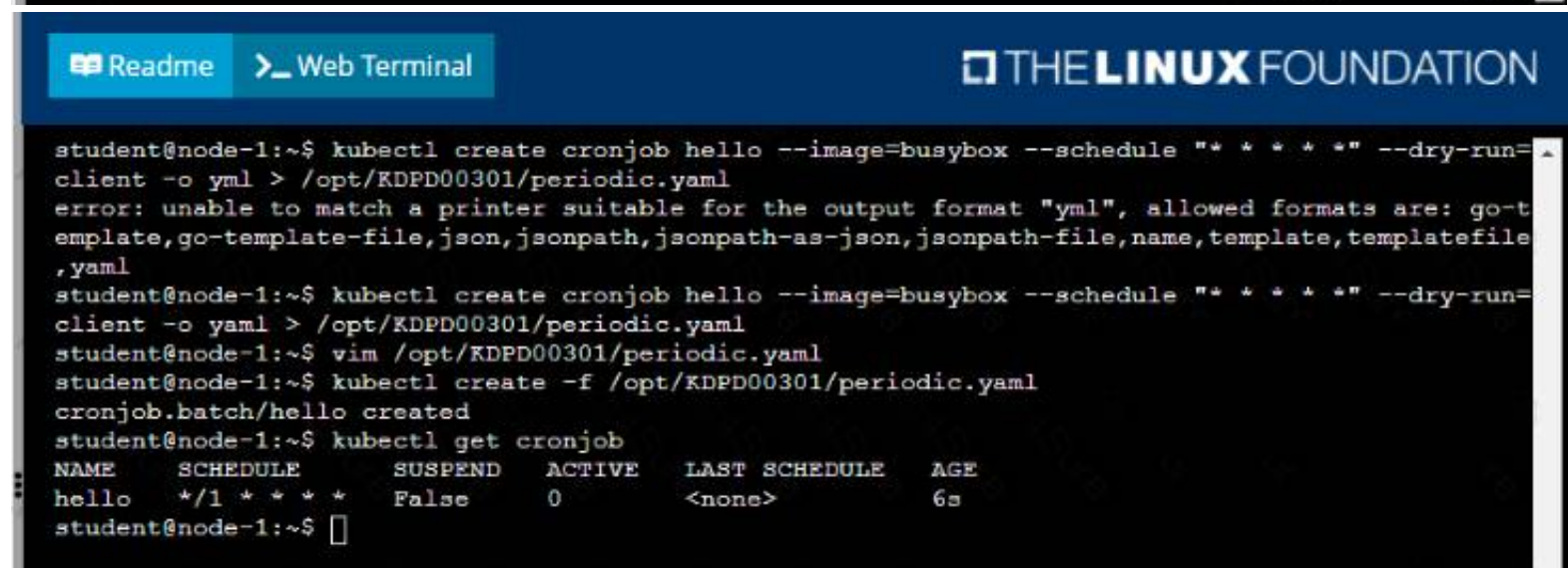A. Mastered
B. Not Mastered

Answer: A

Explanation:
Solution:


```
student@node-1:~$ kubectl create cronjob hello --image=busybox --schedule "* * * * *" --dry-run=
client -o yml > /opt/KDPD00301/periodic.yaml
error: unable to match a printer suitable for the output format "yml", allowed formats are: go-t
emplate,go-template-file,json,jsonpath,jsonpath-as-json,jsonpath-file,name,template,templatefile
,yaml
student@node-1:~$ kubectl create cronjob hello --image=busybox --schedule "* * * * *" --dry-run=
client -o yaml > /opt/KDPD00301/periodic.yaml
student@node-1:~$ vim /opt/KDPD00301/periodic.yaml
```



```
apiVersion: batch/v1beta1
kind: CronJob
metadata:
  name: hello
spec:
  jobTemplate:
    metadata:
      name: hello
    spec:
      template:
        spec:
          containers:
          - image: busybox
            name: hello
            args: ["/bin/sh","-c","date"]
          restartPolicy: Never
  schedule: '*/1 * * * *'
  startingDeadlineSeconds: 22
  concurrencyPolicy: Allow
~
~
~
~
~
~
~
~
                                                              19,26          All
```



```
student@node-1:~$ kubectl create cronjob hello --image=busybox --schedule "* * * * *" --dry-run=
client -o yml > /opt/KDPD00301/periodic.yaml
error: unable to match a printer suitable for the output format "yml", allowed formats are: go-t
emplate,go-template-file,json,jsonpath,jsonpath-as-json,jsonpath-file,name,template,templatefile
,yaml
student@node-1:~$ kubectl create cronjob hello --image=busybox --schedule "* * * * *" --dry-run=
client -o yaml > /opt/KDPD00301/periodic.yaml
student@node-1:~$ vim /opt/KDPD00301/periodic.yaml
student@node-1:~$ kubectl create -f /opt/KDPD00301/periodic.yaml
cronjob.batch/hello created
student@node-1:~$ kubectl get cronjob
NAME    SCHEDULE      SUSPEND   ACTIVE   LAST SCHEDULE   AGE
hello   */1 * * * *   False     0        <none>          6s
student@node-1:~$ []
```

NEW QUESTION 8
Exhibit:

```
Set configuration context:                    ⚠

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

Given a container that writes a log file in format A and a container that converts log files from format A to format B, create a deployment that runs both containers such that the log files from the first container are converted by the second container, emitting logs in format B.
Task:
• Create a deployment named deployment-xyz in the default namespace, that:
•Includes a primary
lfccncf/busybox:1 container, named logger-dev
•includes a sidecar lfccncf/fluentd:v0.12 container, named adapter-zen
•Mounts a shared volume /tmp/log on both containers, which does not persist when the pod is deleted
•Instructs the logger-dev container to run the command

```
while true; do
echo "i luv cncf" >> /
tmp/log/input.log;
sleep 10;
done
```

which should output logs to /tmp/log/input.log in plain text format, with example values:

```
i luv cncf
i luv cncf
i luv cncf
```

• The adapter-zen sidecar container should read /tmp/log/input.log and output the data to /tmp/log/output.* in Fluentd JSON format. Note that no knowledge of Fluentd is required to complete this task: all you will need to achieve this is to create the ConfigMap from the spec file provided at /opt/KDMC00102/fluentd-configma p.yaml , and mount that ConfigMap to /fluentd/etc in the adapter-zen sidecar container

A. Mastered
B. Not Mastered

Answer: A

Explanation:

Solution:

```
📖 Readme   >_ Web Terminal              🔲 THE LINUX FOUNDATION

student@node-1:~$ kubectl create deployment deployment-xyz --image=lfccncf/busybox:1 --dry-run=c
lient -o yaml > deployment_xyz.yml
student@node-1:~$ vim deployment_xyz.yml █
```

```
apiVersion: apps/v1
kind: Deployment
metadata:
  creationTimestamp: null
  labels:
    app: deployment-xyz
  name: deployment-xyz
spec:
  replicas: 1
  selector:
    matchLabels:
      app: deployment-xyz
  strategy: {}
  template:
    metadata:
      creationTimestamp: null
      labels:
        app: deployment-xyz
    spec:
      containers:
      - image: lfccncf/busybox:1
        name: busybox
        resources: {}
status: {}
~
~
~
"deployment_xyz.yml" 24L, 434C                                    3,1          All
```

```
kind: Deployment
metadata:
  labels:
    app: deployment-xyz
  name: deployment-xyz
spec:
  replicas: 1
  selector:
    matchLabels:
      app: deployment-xyz
  template:
    metadata:
      labels:
        app: deployment-xyz
    spec:
      volumes:
      - name: myvol1
        emptyDir: {}
      containers:
      - image: lfccncf/busybox:1
        name: logger-dev
        volumeMounts:
        - name: myvol1
          mountPath: /tmp/log
      - image: lfccncf/fluentd:v0.12
        name: adapter-zen
3 lines yanked                                                   27,22         Bot
```

📖 Readme >_ Web Terminal   THE LINUX FOUNDATION

```
    replicas: 1
    selector:
      matchLabels:
        app: deployment-xyz
    template:
      metadata:
        labels:
          app: deployment-xyz
      spec:
        volumes:
        - name: myvol1
          emptyDir: {}
        containers:
        - image: lfccncf/busybox:1
          name: logger-dev
          command: ["/bin/sh","-c","while [ true ]; do echo 'i luv cncf' >> /tmp/log/input.log; sl
eep 10; done"]
          volumeMounts:
          - name: myvol1
            mountPath: /tmp/log
        - image: lfccncf/fluentd:v0.12
          name: adapter-zen
          command: ["/bin/sh","-c","tail -f /tmp/log/input.log >> /tmp/log/output.log"]
          volumeMounts:
          - name: myvol1
            mountPath: /tmp/log
```

                                                        29,83   Bot

📖 Readme >_ Web Terminal   THE LINUX FOUNDATION

```
    metadata:
      labels:
        app: deployment-xyz
    spec:
      volumes:
      - name: myvol1
        emptyDir: {}
      - name: myvol2
        configMap:
          name: logconf
      containers:
      - image: lfccncf/busybox:1
        name: logger-dev
        command: ["/bin/sh","-c","while [ true ]; do echo 'i luv cncf' >> /tmp/log/input.log; sl
eep 10; done"]
        volumeMounts:
        - name: myvol1
          mountPath: /tmp/log
      - image: lfccncf/fluentd:v0.12
        name: adapter-zen
        command: ["/bin/sh","-c","tail -f /tmp/log/input.log >> /tmp/log/output.log"]
        volumeMounts:
        - name: myvol1
          mountPath: /tmp/log
        - name: myvol2
          mountPath: /fluentd/etc
```

                                                        37,33   Bot

```
student@node-1:~$ kubectl create -f deployment_xyz.yml
deployment.apps/deployment-xyz created
student@node-1:~$ kubectl get deployment
NAME             READY   UP-TO-DATE   AVAILABLE   AGE
deployment-xyz   0/1     1            0           5s
student@node-1:~$ kubectl get deployment
NAME             READY   UP-TO-DATE   AVAILABLE   AGE
deployment-xyz   0/1     1            0           9s
student@node-1:~$ kubectl get deployment
NAME             READY   UP-TO-DATE   AVAILABLE   AGE
deployment-xyz   1/1     1            1           12s
student@node-1:~$ []
```

NEW QUESTION 10
......

Relate Links

100% Pass Your CKAD Exam with Exambible Prep Materials

https://www.exambible.com/CKAD-exam/

Contact us

We are proud of our high-quality customer service, which serves you around the clock 24/7.

Viste - https://www.exambible.com/